

Общий сервер IBM Content Manager OnDemand for
iSeries



Комплект Web-поддержки: Руководство по установке и настройке

Версия 5 Выпуск 3

Общий сервер IBM Content Manager OnDemand for
iSeries



Комплект Web-поддержки: Руководство по установке и настройке

Версия 5 Выпуск 3

Примечание

Перед применением этой информации, а также поддерживаемого ей продукта ознакомьтесь с информацией, приведенной в разделе “Примечания” на стр. 145.

Третье издание (Май 2004)

Это издание предназначено для Общего сервера IBM Content Manager OnDemand for iSeries версия 5, выпуск 3, а также последующих выпусков и модификаций. В новых изданиях об этом будет сообщено дополнительно. Это издание заменяет SC27-1163-01.

© Copyright International Business Machines Corporation 2001, 2004. Все права защищены.

Содержание

О Руководстве по установке и настройке комплекта Web-поддержки Общего сервера IBM Content Manager OnDemand for iSeries. v

Для кого предназначена эта книга	v
Организация этой книги.	v
Предварительные требования и связанная информация	v
Другая информация, доступная в Internet	vi
Навигатор iSeries.	vi
Ждем ваших отзывов	vi

Обзор изменений ix

Глава 1. Обзор. 1

О программных интерфейсах	2
О программах просмотра	4
Работа с программным обеспечением ODWEK	5
Функции продукта	6
Добавить комментарий	6
Изменить пароль	6
Список найденных документов	6
Выход из системы.	6
Вход в систему.	6
Загрузить документ	6
Критерии поиска	7
Печать документа на сервере	7
Обновить документ	7
Просмотреть комментарии.	7
Защита сервера и данных	7

Глава 2. Установка и настройка сервера HTTP 9

Требования для установки	9
Дополнительные требования	9
Установка в системе OS/400	10
Следующий шаг	10
Указание файла ARSWWW.INI	11
[@SRV@_DEFAULT]	11
[@SRV@_server].	12
[CONFIGURATION].	13
[SECURITY]	19
[AFP2HTML]	20
[AFP2PDF].	22
[MIMETYPES]	23
[ATTACHMENT IMAGES].	27
[NO HTML]	29
[DEFAULT BROWSER].	30
[browser]	36
[DEBUG]	37
Пример файла ARSWWW.INI	38
Следующий шаг	40

Глава 3. Настройка примеров приложений 41

LOGON.HTM	41
CREDIT.HTM	41
TEMPLATE.HTM	42
Следующий шаг	43

Глава 4. Установка программ просмотра Web. 45

Обзор	45
Требования	46
Установка	46
Распространение пользовательских файлов	47
Установка файлов AFP Web Viewer.	48
Создание подкаталогов	49
Сохранение пользовательских файлов	49
Настройка файлов шрифтов	50
Создание установочного файла AFP Web Viewer	50
Установка AFP Web Viewer на рабочей станции пользователя	51
Преобразование шрифтов AFP	51
AFP Web viewer	51
Image Web viewer.	52
Программа просмотра строковых данных Java	52
Следующий шаг	57

Приложение А. Справочник по API CGI 59

Добавление комментариев	60
Изменить пароль	63
Список найденных документов	65
Выход из системы	69
Вход в систему	71
Печать документа (Сервер)	73
Загрузить документ	77
Критерии поиска.	80
Обновить документ	83
Просмотреть комментарии	85

Приложение В. Справочник по сервлетам Java. 87

Приложение С. Справочник по API Java 89

Приложение D. Руководство программиста API Java 91

Архитектура клиент-сервер	91
Структура среды Java	91
Советы программисту	92
Настройка среды системы.	92
Установка переменных среды	93
Трассировка и информация диагностики	94
Трассировка	94
Обработка исключительных ситуаций	94

Константы	95
Выполнение приложения ODWEK	95
Подключение к серверу OnDemand	96
Установка соединения	96
Установка и просмотр паролей	96
Работа с сервером OnDemand.	97
Просмотр списка групп приложений папки	99
Поиск папки	100
Поиск в папке с помощью строки SQL	105
Отмена поиска	107
Просмотр списка критериев поиска	109
Просмотр списка папок с информацией о них	112
Просмотр списка документов	114
Загрузка документа	116
Печать документа	119
Просмотр информации о закладках	121
Добавление закладки	123
Обновление документа	124
Изменение пароля	127

Приложение Е. Преобразование AFP в HTML 129

Формат файла AFP2HTML.INI	129
Опции AFP2WEB Transform	130
Просмотр преобразованных документов	131

Приложение F. Преобразование AFP в PDF. 133

Указание файла AFP2PDF.INI	133
Просмотр преобразованных документов	134

Приложение G. файлы конфигурации сервера HTTP 135

HTTP Apache Server	135
Сервер приложений WebSphere	136

Приложение H. Не создавать вывод HTML 137

Вывод ASCII с ограничителями	137
Вход в систему	137
Примечания	138
Критерии поиска	138
Примечания	138
Список найденных документов	139
Примечания	139
Просмотреть комментарии	140
Сообщение об ошибке	140
Примечания	140

| Приложение I. Поддержка языков 141

Приложение J. Инструменты определения неполадок 143

Примечания. 145
Товарные знаки. 147

Индекс 149

О Руководстве по установке и настройке комплекта Web-поддержки Общего сервера IBM Content Manager OnDemand for iSeries

В данной книге приведены инструкции по планированию, установке настройке и применению Комплекта Web-поддержки Общего сервера (OnDemand) IBM Content Manager OnDemand for iSeries версия 5, выпуск 3.

Для кого предназначена эта книга

В основном эта книга предназначена для системных администраторов, которым требуется применить, установить и обслуживать программное обеспечение, а также приложения Комплекта Web-поддержки OnDemand (ODWEK). Кроме того, она может потребоваться программистам, интегрирующим Web-приложения в OnDemand.

Организация этой книги

В этой книге приведена информация, необходимая для установки и настройки программного обеспечения ODWEK, а также планирования прав доступа пользователей к данным Общего сервера IBM Content Manager OnDemand for iSeries с помощью Web-браузера. В состав данной публикации входят следующие разделы:

- Глава 1, “Обзор”, на стр. 1
- Глава 2, “Установка и настройка сервера HTTP”, на стр. 9
- Глава 3, “Настройка примеров приложений”, на стр. 41
- Глава 4, “Установка программ просмотра Web”, на стр. 45
- Приложение А, “Справочник по API CGI”, на стр. 59
- Приложение В, “Справочник по сервлетам Java”, на стр. 87
- Приложение С, “Справочник по API Java”, на стр. 89
- Приложение D, “Руководство программиста API Java”, на стр. 91
- Приложение E, “Преобразование AFP в HTML”, на стр. 129
- Приложение F, “Преобразование AFP в PDF”, на стр. 133
- Приложение G, “файлы конфигурации сервера HTTP”, на стр. 135
- Приложение H, “Не создавать вывод HTML”, на стр. 137
- Приложение I, “Поддержка языков”, на стр. 141
- Приложение J, “Инструменты определения неполадок”, на стр. 143

Предварительные требования и связанная информация

Воспользуйтесь справочной системой IBM iSeries Information Center в качестве отправной точки для поиска технической информации по iSeries.

Доступ к справочной системе Information Center можно получить следующими способами:

- На следующем Web-сайте: <http://www.ibm.com/eserver/iserries/infocenter>
- С компакт-диска, поставляемого вместе с заказом на операционную систему Operating System/400:

iSeries Information Center, SK3T-4091-04. Кроме того, в состав этого пакета входят руководства iSeries в формате PDF, *iSeries Information Center: Дополнительные руководства*, SK3T-4092-01, заменяющие компакт-диск Softcopy Library CD-ROM.

Справочная система Information Center содержит инструкции, а также важные разделы, такие как Java, TCP/IP, Web-обслуживание, защищенные сети, логические разделы, кластеры, команды CL и прикладные программные интерфейсы (API). Кроме того, в ней приведены ссылки на связанные Руководства по выполнению IBM, а также внешние ссылки на другие Web-сайты IBM, такие как Домашняя страница IBM.

Другая информация, доступная в Internet

В Internet доступна дополнительная информация по iSeries. Общую информацию можно получить на Домашней странице iSeries, расположенной по следующему адресу: <http://www-1.ibm.com/servers/eserver/series/>

Описания дополнительных функций iSeries приведены в справочной системе Technical Studio, расположенной по следующему адресу: <http://www.iseries.ibm.com/tstudio/>

Временные исправления программ (PTF) iSeries можно просматривать, выбирать, заказывать и загружать из любой точки мира по Internet. Информация о PTF из Internet (загрузка) iSeries, а также о Планировании профилактического обслуживания (PSP) расположена по следующему адресу: <http://as400service.ibm.com>

Навигатор iSeries

Навигатор iSeries IBM представляет собой мощный графический интерфейс для управления серверами iSeries. В Навигаторе iSeries предусмотрены такие функции, как навигация по системе, настройка, планирование, а также электронная справка, в которой приведены инструкции по выполнению различных задач. Навигатор iSeries позволяет упростить и повысить эффективность работы с сервером и его администрирование. Это единственный пользовательский интерфейс, поддерживающий новые дополнительные функции OS/400. Кроме того, в его состав входит функция Централизованное управление, позволяющая управлять несколькими серверами из центральной системы.

Дополнительная информация о Навигаторе iSeries приведена в справочной системе IBM iSeries Information Center, расположенной по следующему адресу: <http://www.ibm.com/eserver/series/navigator/>

Ждем ваших отзывов

Ваше мнение является важной информацией, позволяющей создать наиболее точную и высококачественную техническую документацию. Отправляйте ваши комментарии о данной публикации и другой документации OnDemand. Отправить комментарии можно следующими способами:

- Отправить по факсу из США, Канады и Пуэрто Рико на номер 1-800-937-3430 (Из других стран: 1-507-253-5192)
- Отправить комментарии по электронной почте по следующему адресу: RCHCLERK@us.ibm.com. Комментарии по справочной системе Information Center отправляйте по адресу RCHINFO@us.ibm.com

Укажите в комментариях следующую информацию:

- Имя книги раздела справочной системы iSeries Information Center

- Номер публикации книги (расположенный в нижнем правом лицевой стороны книги)
- Номер раздела, для которого применяется комментарий

Обзор изменений

В данном издании книги *Руководство по установке и настройке комплекта Web-поддержки Общего сервера IBM Content Manager OnDemand for iSeries* приведена новая и измененная техническая информация. Изменения, внесенные в некоторые экземпляры, могли быть не обозначены с помощью маркеров изменения. Следует отметить следующие изменения:

- В продукте Content Manager OnDemand for iSeries (OnDemand) версия 5, выпуск 1 представлена новая реализация сервера под названием Общий сервер. Общий сервер предоставляет пользователям и администраторам расширенные возможности по индексации, просмотру, защите, работе с форматом PDF, а также Web-поддержки. Текущие пользователи OnDemand, работающие с архивом буферных файлов (вместе с AnyStore без нее или с существующей функцией Сервер), могут перейти к новой версии с Общим сервером следуя инструкциям, приведенным в Приложении А Руководства по установке и настройке Общего сервера Content Manager OnDemand for iSeries. Обратите внимание на то, что в документации преобразование Архива буферных файлов включает в себя также описание преобразования данных AnyStore, если AnyStore установлена.
- В публикацию Общий сервер Content Manager OnDemand for iSeries: Руководство по индексации были внесены значительные дополнения, касающиеся функций, поддерживаемых Индексатором OS/400. Среди этих дополнений разделы, связанные с определением многоключевых индексов, полей транзакций, полей текстового поиска, буферных файлов SCS с перекрытиями AFP, а также масок для полей приложений.
- В продукте Content Manager OnDemand for iSeries реализована поддержка новых оптических библиотек Plasmon, поддерживаемых iSeries.
- Для упрощения работы с командой Запустить архивированное управление памятью OnDemand (STRASMOND) был удалены два ее параметра. Более подробная информация приведена в Приложении А Руководства администратора Общего сервера Content Manager OnDemand for iSeries.
- Поддержка стандартного сервера HTTP отменена в OS/400. В связи с этим также поддержка стандартного сервера HTTP удалена из Комплекта Web-поддержки Content Manager OnDemand (ODWEK). Программное обеспечение ODWEK теперь поддерживает только сервер HTTP Apache.

Глава 1. Обзор

Программное обеспечение ODWEK позволяет пользователям получать доступ к данным сервера OnDemand IBM Content Manager с помощью Web-браузера или пользовательской программы. Например, пользователям можно предоставить URL Web-страницы, позволяющей войти на сервер OnDemand. Кроме того, можно создать URL Web-страниц для поиска в заданной папке. ODWEK проверяет правильность информации о пользователе, такой как права доступа и данные группы приложений, на сервере OnDemand. После отправки запроса на поиск ODWEK открывает Web-страницу со списком соответствующих документов. Пользователь выбирает необходимый документ для просмотра и ODWEK открывает его в браузере.

На рисунке рис. 1 показан Web-браузер рабочей станции, с помощью которого получен доступ к данным сервера OnDemand.

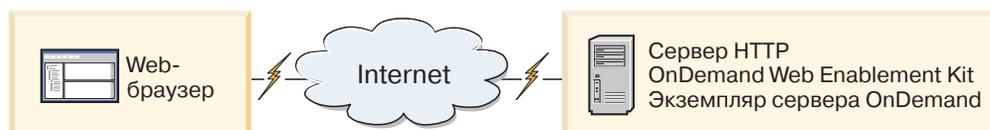


Рисунок 1. Обращение к данным сервера OnDemand с помощью ODWEK

Программа ODWEK позволяет загружать найденные документы с серверов OnDemand, в которых выполняется Общий сервер IBM Content Manager OnDemand for iSeries версии 5, IBM Content Manager OnDemand for Multiplatforms версии 7 и IBM Content Manager OnDemand for z/OS и OS/390 версии 7.1.

В состав программного обеспечения ODWEK входят следующие компоненты:

- Программный интерфейс OnDemand. Программный интерфейс использует стандартные интерфейсы и протоколы OnDemand для обращения к данным сервера OnDemand. Дополнительный исходный код для поддержки ODWEK на сервере OnDemand не требуется. Для управления ODWEK предусмотрены следующие программные интерфейсы:
 - Программа Common Gateway Interface (CGI). Программа CGI позволяет получать доступ к данным OnDemand с помощью Web-браузера. Программа CGI работает в системах, в которых выполняется сервер HTTP, такой как IBM HTTP Server.
 - Сервлет Java. Программа CGI позволяет получать доступ к данным OnDemand с помощью Web-браузера. Сервлет работает на сервере HTTP с поддержкой Java, в котором выполняется сервер приложений Java, такой как IBM WebSphere Application Server.
 - API Java. API Java позволяет обращаться к данным OnDemand с помощью пользовательской программы. Для работы API Java требуется Java версии 1.2.2 или более поздней.
- IBM OnDemand Advanced Function Presentation (AFP) Web Viewer. Программа AFP Web Viewer позволяет искать, загружать, просматривать, печатать документы AFP и перемещаться по ним с помощью Web-браузера.
- Программа IBM OnDemand Image Web Viewer. Программа Image Web Viewer позволяет пользователям искать, просматривать, печатать документы BMP, GIF, JPEG, PCX и TIFF, а также перемещаться по ним с помощью Web-браузера.

- Апплет Строковые данные Java. Апплет Строковые данные позволяет просматривать документы со строковыми данными с помощью Web-браузера. Для применения апплета Строковые данные администратор должен настроить файл ARSWWW.INI.
- Апплет AFP2HTML Java. Апплет AFP2HTML позволяет пользователям просматривать вывод, созданный служебной программой IBM AFP2WEB Transform. Служебная программа AFP2WEB Transform преобразует документы и ресурсы AFP в файлы HTML, которые можно просмотреть с помощью апплета AFP2HTML. После установки и настройки служебной программы AFP2WEB Transform для применения апплета AFP2HTML администратор должен настроить файл ARSWWW.INI.

Примечание: Для просмотра документов других типов на сервере OnDemand следует загрузить и установить соответствующие программы просмотра. Например, для просмотра документов в формате PDF IBM рекомендует загрузить программу просмотра Adobe Acrobat для браузеров, применяемых в вашей компании.

О программных интерфейсах

Экземпляр ODWEK представляет собой исходный код ODWEK, обращающийся к данным сервера OnDemand. Экземпляр управляет правами доступа к данным, а также управляет ресурсами системы, присвоенными этим данным. Каждый экземпляр является независимой средой. Для каждого экземпляра предусмотрены отдельный файл ASWWW.INI и программный интерфейс ODWEK, недоступные другим экземплярам. Предусмотрены следующие три программных интерфейса ODWEK:

- Программа CGI - интерфейс между Web-браузером и сервером OnDemand
- Сервлет Java - интерфейс между Web-браузером и сервером OnDemand
- API Java - набор методов, позволяющих обращаться к данным OnDemand с помощью пользовательских программ

Обратите внимание, что каждый экземпляр может использовать только один программный интерфейс. Программные интерфейсы взаимно исключают друг друга. Их нельзя одновременно применять в одном экземпляре. Однако, несколько экземпляров ODWEK могут выполняться в одной системе, если они применяют разные программные интерфейсы. Для этого для каждого экземпляра следует указать отдельный номер порта.

Наиболее распространен способ реализации ODWEK, когда в системе выполняется один экземпляр. Конфигурация с одним экземпляром типична для автономных вычислений и вычислений, связанных с разработкой, для которых требуется один экземпляр сервера приложений, работающий независимо от остальных приложений.

На рисунке рис. 2 показан пример отдельного экземпляра с интерфейсом CGI.

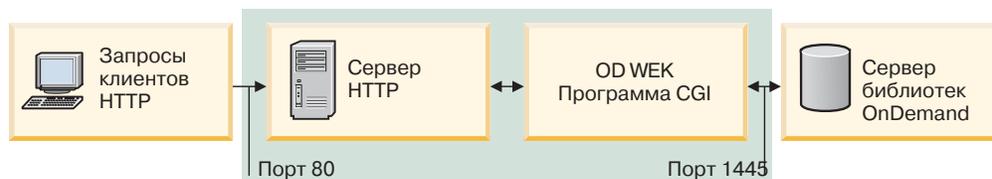


Рисунок 2. Один экземпляр с интерфейсом CGI

На рисунке рис. 3 приведен пример отдельного экземпляра с интерфейсом сервлета Java.

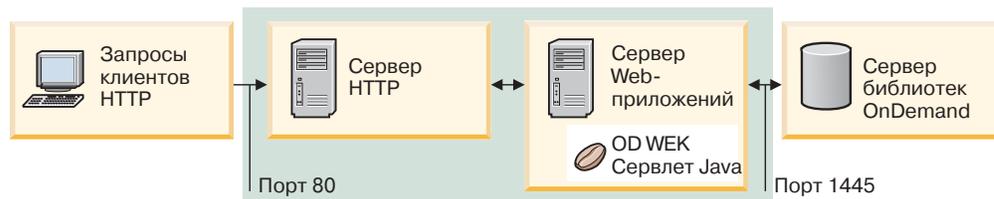


Рисунок 3. Один экземпляр с интерфейсом Java

На рисунке рис. 4 показан пример отдельного экземпляра с интерфейсом API Java.



Рисунок 4. Один экземпляр с интерфейсом API Java

В одной системе можно настроить несколько экземпляров ODWEK. Для каждого экземпляра требуется отдельный программный интерфейс и файл ARSWWW.INI, в котором указан уникальный номер порта для обмена данными между программным интерфейсом и сервером OnDemand. Кроме того, для каждого экземпляра необходима собственная память и защита. Конфигурация с несколькими экземплярами обычно применяется, если в одной системе должно выполняться несколько приложений разработки, тестирования или производства. Экземпляры работают независимо друг от друга.

На рисунке рис. 5 на стр. 4 приведен пример топологии с несколькими экземплярами.

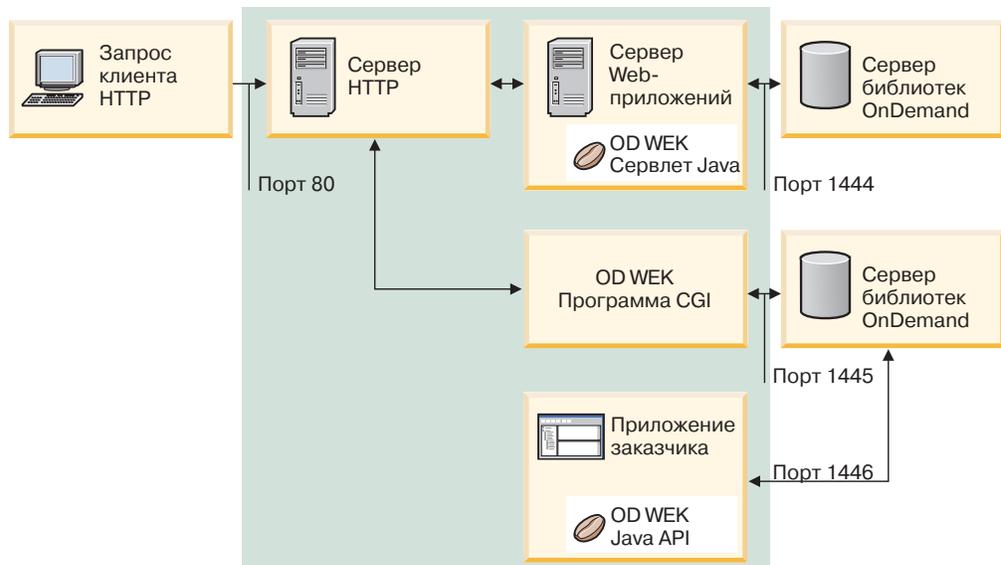


Рисунок 5. Топология с несколькими экземплярами

О программах просмотра

В программном обеспечении ODWEK предусмотрены следующие программы просмотра:

- AFP Web Viewer
- Image Web Viewer
- Апплет Строковые данные Java
- Апплет AFP2HTML Java

Программы AFP Web Viewer и Image Web Viewer расширяют возможности Web-браузера. Программа AFP Web Viewer позволяет просматривать документы AFP. Программа Image Viewer позволяет просматривать документы BMP, GIF, JPEG, PCX и TIFF. Программы просмотра предназначены для просмотра документов в окне браузера. Каждая программа просмотра добавляет панель инструментов в верхнюю часть окна просмотра. Панель инструментов может расширять панель инструментов браузера. Панель инструментов встраиваемого модуля позволяет управлять вспомогательными функциями, необходимыми для работы с документами. Для работы с программами просмотра их необходимо установить на рабочие станции пользователей.

Примечание: Программы просмотра устанавливаются в качестве встраиваемых модулей или компонентов ActiveX. Если на рабочей станции установлен Internet Explorer, то устанавливаются компоненты ActiveX. Если установлен Netscape, устанавливаются встраиваемые модули. Если на рабочей станции установлен как Internet Explorer, так и Netscape, то устанавливаются компоненты ActiveX для Internet Explorer и встраиваемые модули для Netscape.

Апплет Строковые данные позволяет просматривать документы SCS и строковые данные, расположенные на сервере OnDemand. Апплет Строковые данные отображает документы строковых данных в окне браузера и добавляет панель инструментов в верхнюю часть экрана. Панель инструментов апплета Строковые данные позволяет

управлять вспомогательными функциями, необходимыми для работы с документами. Для применения апплета Строковые данные администратор должен настроить файл ARSWWW.INI.

Апплет AFP2HTML позволяет пользователям просматривать вывод, созданный служебной программой IBM AFP2WEB Transform. Служебная программа AFP2WEB Transform преобразует документы и ресурсы AFP в документы HTML. После установки и настройки служебной программы AFP2WEB Transform для применения апплета AFP2HTML администратор должен настроить файл ARSWWW.INI. Апплет AFP2HTML добавляет панель инструментов, в которой предусмотрены вспомогательные функции, необходимые для работы с документами, включая работу с большими объектами.

Основным преимуществом апплетов является то, что, в отличие от программ просмотра, пользователям не требуется устанавливать программное обеспечение на рабочие станции и обновлять его. Кроме того, если компания IBM предоставит новую версию программы просмотра, ее потребуется отправить всем пользователям.

При работе с апплетами и программами просмотра, разработанными компанией IBM документы, загружаются с сервера OnDemand на компьютер клиента в сжатом виде. Клиент распаковывает документы и отображает страницы в окне Web-браузера. Если документ сохранен на сервере OnDemand в качестве большого объекта, то клиент загружает и распаковывает отдельные фрагменты документа по мере просмотра страниц документа пользователем.

Работа с программным обеспечением ODWEK

Для работы с программным обеспечением ODWEK рекомендуется настроить примеры приложений HTML, поставляемых вместе с ним, в соответствии с конкретными потребностями. Пример приложения LOGON.HTM предназначен для пользователей, которым разрешен доступ к нескольким папкам. Сначала следует указать на странице LOGON.HTM информацию о сервере OnDemand. Затем опубликовать URL файла LOGON.HTM. С помощью этого URL пользователи смогут войти на указанный сервер. Программа ODWEK автоматически отображает набор Web-страниц, с помощью которых можно искать, загружать и просматривать документы OnDemand. Пример приложения CREDIT.HTM предназначен для нерегулярной работы с OnDemand и предоставляет Web-страницу с критериями поиска для определенной папки. После настройки примера пользователь может перейти по указанному URL, задать критерии поиска и нажать кнопку Отправить. ODWEK вернет Web-страницу, в которой перечислены документы, возвращенные в результате запроса.

Важная информация: В процессе работы ODWEK записывает cookie на компьютер клиента. Убедитесь, что браузеры пользователей настроены таким образом, что cookie принимаются.

Большинство заказчиков определяют один ИД пользователя OnDemand для входа на сервер ODWEK. Такой подход рекомендуется в средах, в которых к одной и той же папке будет обращаться большое число пользователей. Кроме того, вы можете создать ИД пользователя OnDemand для каждого пользователя. ИД пользователей необходимо управлять независимо от выбора способа предоставления прав доступа к серверу OnDemand с ODWEK: их следует добавить на сервер, а также задать группу приложений и права доступа пользователей к папке.

Функции продукта

ODWEK поддерживает следующие функции OnDemand. Как правило, функции вызываются с помощью Web-страниц, содержащих ссылки на программы сервера ODWEK. Каждая ссылка вызывает конкретную функцию. Вывод функции представляет собой отдельную Web-страницу, в которой приведены ссылки на следующую логическую функцию. Например, начальная Web-страница может вызывать функцию входа в систему. Функция Вход в систему создает Web-страницу с ссылкой на функцию Критерии поиска. Для вызова функций применяется прикладной программный интерфейс (API). Дополнительная информация приведена в разделе Приложение A, “Справочник по API CGI”, на стр. 59.

Добавить комментарий

Функция Добавить комментарий позволяет добавлять комментарии в указанный документ. Для того чтобы добавить комментарий пользователь должен обладать правами доступа на добавление комментариев для каждой группы приложений, содержащих необходимые документы. (Права доступа к группе приложений позволяют добавлять комментарии.)

Изменить пароль

Функция Изменить пароль позволяет пользователям изменять свои пароли OnDemand.

Список найденных документов

Функция Список найденных документов позволяет создать список элементов, соответствующих критериям поиска. Данный список отображается в виде таблицы HTML. Каждый элемент, соответствующий критериям поиска, сохраняется в виде отдельной строки таблицы и содержит ссылку на функцию Загрузить документ.

| **Важное примечание для заказчиков, применяющих как архив буферных файлов**
| **OnDemand, так и общий сервер, а также запись ARS_MIGR_SERVER в файле**
| **ARS.CFG для совместного использования папок архива буферных файлов и общего**
| **сервера в списке выбора одной папки:** Для пользователей ODWEK папки архива
| буферных файлов отображаются в папке ODWEK и доступны для поиска. Однако при
| этом нельзя загрузить документ.

Выход из системы

Функция Выход из системы позволяет выйти из сервера OnDemand.

Вход в систему

Функция Вход в систему позволяет пользователям войти на сервер OnDemand. В случае успешного выполнения функции Вход в систему отображается Web-страница, в которой приведен список папок, доступных для пользователя.

Загрузить документ

Функция Загрузить документ позволяет загрузить документ с сервера OnDemand. Поток данных, возвращаемый сервером включает в себя сам документ, а также, в зависимости от типа документа, ресурсы, необходимые для его просмотра. Поток данных не следует изменять. Браузер, совместно с программой просмотра, расшифровывает поток данных и отображает документ. Если документ сохранен на сервере OnDemand в качестве большого объекта, то возвращается только первый фрагмент документа. Последующие фрагменты загружаются и отображаются по мере необходимости.

Критерии поиска

В случае успешного входа в систему отображается Web-страница, в которой приведен список папок, доступных для пользователя. Пользователь должен выбрать необходимую папку. При этом отображается Web-страница содержащая поля поиска, предусмотренные для папки. Пользователь может выбрать критерии поиска по умолчанию или ввести критерии для поиска конкретных документов. При нажатии кнопки Отправить запрос на поиск передается на сервер OnDemand.

Печать документа на сервере

Функция Печать документа на сервере позволяет отправить копии документов на принтер сервера OnDemand. Для того чтобы распечатать документы пользователь должен обладать правами доступа на печать документов для каждой группы приложений, в которых содержатся необходимые документы. (Права доступа к группе приложений позволяют распечатать документ.) На сервере OnDemand должен быть определен принтер сервера.

Обновить документ

Функция Обновить документ позволяет обновить записи базы данных. Функция Обновить документ обновляет одно или несколько полей указанного документа.

Просмотреть комментарии

Функция Просмотреть комментарии позволяет просмотреть комментарии, добавленные к указанному документу. Для того чтобы просмотреть комментарии пользователь должен обладать правами доступа на просмотр комментариев для каждой группы приложений, в которых содержатся необходимые документы. (Права доступа к группе приложений позволяют просматривать комментарии.)

Защита сервера и данных

Перед работой с ODWEK рассмотрите следующие уровни защиты:

- Доступ к программам ODWEK и Web-страницам
- Доступ к данным сервера OnDemand

Любой пользователь, обладающий доступом к серверу HTTP, программам и Web-страницам входящим в состав комплекта ODWEK потенциально может получить доступ к данным сервера OnDemand. IBM настоятельно рекомендует ограничивать доступ к этим программам и Web-страницам. Ограничить доступ к программам и Web-страницам сервера HTTP можно различными способами. Например, во многих серверах HTTP предусмотрена система защиты важных Web-страниц, позволяющая ограничить доступ к каталогам. Кроме того, сервер HTTP можно защитить с помощью файла паролей. При этом для получения доступа к Web-страницам пользователям потребуется ввести ИД пользователя и пароль. Однако не смотря на то, что ИД пользователей и пароли сервера HTTP и операционной системы аналогичны, между ними не существует соответствия. Кроме того, соответствия не существует между ИД пользователей и паролями сервера HTTP и сервера OnDemand.

Программа ODWEK предоставляет возможность доступа к данным сервера OnDemand с помощью стандартных API OnDemand. API проверяют права доступа ИД пользователя OnDemand к запрашиваемым данным сервера. В организации должен быть сотрудник, отвечающий за администрирование пользователей и защиту данных сервера OnDemand.

Рекомендуется обратить внимание на еще одну особенность, связанную с защитой: метод передачи параметров формы между клиентом и сервером. В формах, поставляемых вместе с комплектом ODWEK, применяется метод POST, позволяющий передавать параметры и значения в теле запроса HTTP. При работе с методом POST параметры и значения не отображаются в поле Адрес Web-браузера. Например, вызов функции может выглядеть следующим образом:

```
http://www.company.com/cgi-bin/arswww.cgi
```

Однако, если при создании формы метод не указан, то используется метод по умолчанию GET, который передает параметры и значения в URL. Вызов функции с помощью метода GET может выглядеть следующим образом:

```
http://www.company.com/cgi-bin/arswww.cgi?_function=logon  
&_user=bob&_password=secret
```

Параметры и значения отображаются явным образом в поле Адрес Web-браузера. При создании форм IBM настоятельно рекомендует применять метод POST. Для изменения метода по умолчанию с GET на POST следует указать атрибут METHOD в теге form.

Примечание: Если метод GET все-таки необходим, вы можете зашифровать параметры и значения. Для этого укажите параметр ENCRYPTURL в файле ARSWWW.INI. За дополнительной информацией обратитесь к разделу “ENCRYPTURL” на стр. 33.

Глава 2. Установка и настройка сервера HTTP

В этом разделе приведены требования для установки программного обеспечения ODWEK, инструкции по его установке на сервер HTTP, а также инструкции по изменению файла конфигурации ODWEK.

Программное обеспечение ODWEK следует устанавливать в системе iSeries, в которой установлена текущая версия IBM HTTP Server. Кроме того, если вы планируете работать с сервлетом Java, убедитесь, что установлена и выполняется текущая версия iSeries Web Application Server (WebSphere).

Программное обеспечение ODWEK может выполнять поиск документов и загружать их с серверов OnDemand, работающих под управлением Общего сервера IBM Content Manager OnDemand for iSeries версии 5, выпуска 3.

Требования для установки

Требования для установки программного обеспечения ODWEK:

- Текущая версия IBM HTTP Apache Server. Кроме того, если вы планируете работать с сервлетом Java, убедитесь, что установлена и выполняется текущая версия iSeries Web Application Server (WebSphere). Серверы должны работать под управлением OS/400 версии 5, выпуска 3.
- Накопитель соответствующего типа для установки.
- Достаточный объем дисковой памяти для установки: примерно 30 Мб для сервера HTTP.
- Достаточный объем дисковой памяти для хранения кэша: 10 Мб по умолчанию для сервера HTTP. За дополнительной информацией обратитесь к разделу “CACHESIZE” на стр. 15.

Дополнительные требования

Программное обеспечение ODWEK позволяет *кэшировать* (временно сохранять в памяти) документы на сервере HTTP. Такой подход позволяет увеличить скорость отправки уже просмотренных документов пользователям. Для того чтобы разрешить кэширование документов необходимо настроить параметр CACHEDOCS в файле ARSWWW.INI. Дополнительная информация приведена в разделе “CACHEDOCS” на стр. 14.

По умолчанию кэш-память ODWEK расположена в каталоге /QIBM/UserData/OnDemand/WWW/CACHE. Для того чтобы изменить расположение кэш-памяти следует внести изменения в файл ARSWWW.INI. Дополнительная информация приведена в разделе “CACHEDIR” на стр. 14.

Убедитесь, что процессы, запускающие программы ODWEK имеют права доступа на чтение из каталогов этих программ и на запись в каталог кэш-памяти. В процессе установки программного обеспечения ODWEK все объекты защищаются с помощью списка прав доступа QONDADM, в который добавляются пользовательские профайлы QTMNHTTP, QTMNHTP1 и QEJBSVR с правами доступа *CHANGE. Кроме того, в список прав доступа QRDARS400 должны быть добавлены пользовательские профайлы QTMNHTTP, QTMNHTP1 и QEJBSVR с правами доступа *USE.

Если вы планируете работать с апплетом AFP2HTML, получите служебную программу AFP2WEB Transform из IBM, установите и настройте ее на сервере HTTP. Более подробную информацию о служебной программе AFP2WEB Transform можно получить в представительстве IBM. Кроме того, для документов AFP и ресурсов, обрабатываемых с помощью AFP2WEB Transform, необходимо указать опции конфигурации. Дополнительная информация о файле конфигурации приведена в разделе Приложение E, “Преобразование AFP в HTML”, на стр. 129.

Если вы планируете работать преобразовывать документы AFP, расположенные на сервере OnDemand в формат PDF для просмотра с помощью программы Adobe Acrobat, получите служебную программу AFP2PDF Transform из IBM, установите и настройте ее на сервере HTTP. Более подробную информацию о служебной программе AFP2PDF Transform можно получить в представительстве IBM. Кроме того, для документов AFP и ресурсов, обрабатываемых с помощью AFP2PDF Transform, необходимо указать опции конфигурации. Дополнительная информация о файле конфигурации приведена в разделе Приложение F, “Преобразование AFP в PDF”, на стр. 133. Для просмотра преобразованных документов необходимо получить программу просмотра Adobe Acrobat для браузеров, применяемых в вашей организации.

Установка в системе OS/400

Для установки программного обеспечения ODWEK выполните следующие действия:

1. Загрузите последнюю копию файла README OnDemand. Распечатайте его и внимательно прочтите, перед началом установки.
2. Для установки ODWEK выполните инструкции, приведенные в книге *Установка программ* (SC41-5120). Номер лицензионной программы 5722RD1, номер функции - 11.

Примечание: Рекомендуется устанавливать ODWEK с помощью опции Установить лицензионные программы меню Работа с лицензионными программами (licpgm). В меню Установить лицензионные программы введите 1 для добавления опции, введите номер лицензионной программы 5722RD1 и номер функции 11, либо найдите ODWEK в списке Лицензионные программы и Опции продукта и укажите 1 напротив этого элемента. **В случае установки OnDemand с помощью других методов могут произойти ошибки.**

3. IBM рекомендует после успешной установки лицензионной программы заказать, загрузить и установить PTF, доступные для OnDemand. Полный список PTF, предусмотренных для OnDemand версии 3, выпуска 3 приведен в информационном APAR П13680. Информационный APAR в электронном виде можно заказать с помощью команды SNDPTFORD, указав в качестве номера PTF значение П13680. Ознакомьтесь с описанием PTF и выполните все специальные инструкции.

Следующий шаг

Убедитесь, что текущая версия IBM HTTP Server установлена в системе iSeries и выполняется. Для сервера HTTP потребуется дополнительная настройка. Пример файла конфигурации сервера HTTP приведен в разделе Приложение G, “файлы конфигурации сервера HTTP”, на стр. 135.

Если вы планируете работать с сервлетом Java, убедитесь, что установлена и выполняется текущая версия iSeries Web Application Server (WebSphere). Для сервера WebSphere потребуется дополнительная настройка. Инструкции приведены в справочной системе IBM WebSphere Application Server for AS/400 Documentation Center, расположенной в Internet по адресу

www.ibm.com/servers/eserver/series/software/websphere/wsappserver/. Для установки и начальной настройки WebSphere перейдите по ссылкам для соответствующей версии.

После установки программного обеспечения ODWEK, настройки сервера HTTP и настройки WebSphere (при необходимости) настройте файл инициализации ODWEK в соответствии с особенностями операционной системы. См. “Указание файла ARSWWW.INI”.

Указание файла ARSWWW.INI

Файл ARSWWW.INI представляет собой текстовый файл ASCII, содержащий параметры, необходимые для программ ODWEK (таких как программа CGI или серверлет Java). Каждый параметр указывается в отдельной строке. При этом применяется следующий формат: ПАРАМЕТР=значение. Например:

```
AFPVIEWING=plugin
CACHEDIR=/tmp/cache
LANGUAGE=ENU
```

Параметры файла ARSWWW.INI расположены в разделах. Начало раздела указывается с помощью заголовка раздела в следующем формате: [заголовокРаздела]. После заголовка раздела укажите параметры. Например:

```
[@SRV@_gunnar]
HOST=gunnar
PORT=1446
PROTOCOL=0
```

Пример файла конфигурации ARSWWW.INI поставляется вместе с продуктом. В примере файла конфигурации приведен набор наиболее часто используемых значений. Пример находится в разделе “Пример файла ARSWWW.INI” на стр. 38.

Для файла ARSWWW.INI предусмотрены следующие разделы и параметры:

[@SRV@_DEFAULT]

Раздел сервера по умолчанию. Раздел сервера по умолчанию позволяет указать общие параметры серверов OnDemand, обменивающихся данными с программным обеспечением ODWEK. Параметры и значения, указанные в этом разделе, применяются только в том случае, если они не указаны в разделе сервера.

Область действия этого раздела распространяется на все серверы. В файле ARSWWW.INI ее следует указывать только один раз.

Это необязательный раздел.

В данном разделе предусмотрены следующие параметры:

PORT

Номер порта TCP/IP, применяемого серверами OnDemand для связи с ODWEK. Если параметр PORT не указан, то применяется номер порта, заданный для OnDemand в таблице служб (WRKSRVTBLE). Если параметр PORT не указан и OnDemand не содержится в таблице служб, то по умолчанию применяется порт 1445. Для применения номера порта, заданного для OnDemand в таблице служб, укажите значение 0 (ноль).

Данный параметр в разделе по умолчанию можно указывать только один раз. В случае применения API Вход в систему указанный номер порта можно переопределить с помощью параметра `_port`.

Это необязательный параметр.

Пример:

```
[@SRV@_DEFAULT]  
PORT=0
```

PROTOCOL

Сетевой протокол, применяемый серверами OnDemand для связи с ODWEK. В случае применения TCP/IP укажите 0 (ноль).

Данный параметр в разделе по умолчанию можно указывать только один раз.

Это необязательный параметр. Если этот параметр не указан, то по умолчанию применяется значение 0 (ноль).

Пример:

```
[@SRV@_DEFAULT]  
PROTOCOL=0
```

[@SRV@_server]

Раздел сервера. Раздел сервера следует указать для каждого сервера, обменивающегося данными с ODWEK. Раздел сервера содержит параметры и значения, применяемые для конкретного сервера. Заголовок раздела должен содержать строку, обозначающую сервер. Параметры, указанные в разделе сервера, переопределяют параметры из раздела по умолчанию.

Для каждого сервера следует указать отдельный раздел сервера.

Это обязательный раздел.

В данном разделе предусмотрены следующие параметры:

HOST

Имя сервера OnDemand. Вы можете указать адрес TCP/IP, псевдоним или полное имя хоста сервера.

Данный параметр в разделе сервера следует указывать только один раз.

Это обязательный параметр.

Пример:

```
[@SRV@_gunnar]  
HOST=gunnar
```

PORT

Номер порта TCP/IP, применяемого сервером OnDemand для связи с ODWEK. Если параметр PORT не указан, то применяется номер порта, заданный в разделе по умолчанию.

Данный параметр в разделе сервера можно указывать только один раз. В случае применения API Вход в систему указанный номер порта можно переопределить с помощью параметра `_port`.

Это необязательный параметр.

Пример:

```
[@SRV@_gunnar]
PORT=0
```

Этот номер порта должен совпадать с номером, указанным в файле `ars.ini` для экземпляра.

PROTOCOL

Сетевой протокол, применяемый сервером OnDemand для связи с ODWEK. В случае применения TCP/IP укажите 0 (ноль).

Данный параметр в разделе сервера можно указывать только один раз.

Это необязательный параметр. Если значение не указано, то применяется значение, заданное в разделе по умолчанию.

Пример:

```
[@SRV@_gunnar]
PROTOCOL=0
```

[CONFIGURATION]

В разделе CONFIGURATION содержатся параметры, применяемые ODWEK для работы с сервером HTTP.

Область действия этого раздела является глобальной. В файле `ARSWWW.INI` ее следует указывать только один раз.

Это необязательный раздел.

В данном разделе предусмотрены следующие параметры:

APPLETCACHEDIR

Задаёт каталог временного хранения документов для апплетов Строковые данные и AFP2HTML. Вы можете указать локальный каталог или каталог сетевого диска. Все пользователи должны обладать правами доступа на запись в этот каталог.

Пример:

```
[Configuration]
APPLETCACHEDIR=/QIBM/UserData/OnDemand/www/cache
```

Примечания:

1. Область действия параметра APPLETCACHEDIR является глобальной.
2. Параметр APPLETCACHEDIR необязателен. Однако, если он не указан, то апплеты будут пытаться сохранять документы с рабочем каталоге Java.
3. Если указанный каталог не существует, то апплеты будут сохранять документы в рабочем каталоге Java.
4. Апплет удаляет документы из каталога кэш-памяти при завершении работы этого апплета (например, при закрытии документа).

APPLETDIR

Задаёт каталог, в котором расположены апплеты Строковые данные и AFP2HTML.

Примечания:

1. Вы можете указать имя каталога или AliasMatch:
 - Имя каталога следует указывать относительно каталога `/QIBM/UserData/OnDemand/WWW`. Например, если указан параметр

| appletdir=applets, то апплеты должны существовать в каталоге
| /QIBM/UserData/OnDemand/WWW/APPLETS.

- AliasMatch должен быть определен в файле конфигурации сервера HTTP. Например, если указан параметр appletdir=/applets/, то в файле конфигурации сервера HTTP должен быть указан AliasMatch для /applets/. В правиле AliasMatch в качестве заменяемого пути к файлу должен быть указан полный путь к каталогу сервера. Например:

| AliasMatch ^/applets/com/ibm/edmslod/(.*)\$ /QIBM/UserData/OnDemand/www/applets/\$1

| AliasMatch ^/applets/(.*)\$ /QIBM/UserData/OnDemand/www/applets/\$1

2. Проверьте права доступа к указанному каталогу. Процессы, запускающие программы ODWEK, должны иметь права доступа к каталогу апплетов.

Область действия этого параметра является глобальной. В разделе CONFIGURATION его следует указывать только один раз.

Это обязательный параметр.

Пример:

```
[CONFIGURATION]  
APPLETDIR=applets
```

CACHEDIR

Позволяет указать каталог сервера HTTP для временного хранения (*кэширования*) документов ODWEK (см. "CACHEDOCS"). По умолчанию кэш-память ODWEK расположена в каталоге /QIBM/UserData/OnDemand/WWW/CACHE .

Примечание: Проверьте права доступа к указанному каталогу. Процессы, запускающие программы ODWEK, должны иметь права доступа к каталогу кэш-памяти на чтение и запись.

Область действия этого параметра является глобальной. В разделе CONFIGURATION его следует указывать только один раз.

Это необязательный параметр.

Пример:

```
[CONFIGURATION]  
CACHEDIR=/QIBM/UserData/OnDemand/WWW/CACHE
```

CACHEDOCS

Определяет, следует ли ODWEK временно сохранять (*кэшировать*) документы на сервере HTTP. Применение кэш-памяти позволяет увеличить скорость загрузки уже просмотренных документов с сервера. Значение по умолчанию - 0 (ноль). Оно означает, что кэширование документов не применяется. Для того чтобы разрешить кэширование документов, укажите 1. В этом случае проверьте каталог кэширования документов ODWEK (см. "CACHEDIR"), а также объем доступной для этой цели дисковой памяти (см. "CACHESIZE" на стр. 15).

Примечание: IBM рекомендует всегда применять кэширование документов при работе с браузером Internet Explorer Microsoft, а также программами просмотра AFP Web Viewer и Image Web Viewer.

Область действия этого параметра является глобальной. В разделе CONFIGURATION его следует указывать только один раз.

Это необязательный параметр. Однако в большинстве случаев рекомендуется настраивать кэш-память для документов.

Пример:

```
[CONFIGURATION]  
CACHEDOCS=1
```

CACHEMAXTHRESHOLD

Определяет момент начала удаления данных и документов ODWEK из кэш-памяти. Удаление данных и документов ODWEK начинается, если процент используемой дисковой памяти превышает указанное значение. Значение по умолчанию - 80 (восемьдесят процентов). ODWEK удаляет наиболее старые элементы кэш-памяти до тех пор, пока не будет достигнуто пороговое значение (см. "CACHEMINTHRESHOLD").

Область действия этого параметра является глобальной. В разделе CONFIGURATION его следует указывать только один раз.

Это необязательный параметр.

Пример:

```
[CONFIGURATION]  
CACHEMAXTHRESHOLD=80
```

CACHEMINTHRESHOLD

Определяет момент прекращения удаления данных и документов ODWEK из кэш-памяти. Удаление данных и документов ODWEK прекращается, если процент используемой дисковой памяти меньше указанного значения. Значение по умолчанию - 40 (сорок процентов). ODWEK начинает удалять наиболее старые элементы кэш-памяти при достижении порогового значения (см. "CACHEMAXTHRESHOLD").

Область действия этого параметра является глобальной. В разделе CONFIGURATION его следует указывать только один раз.

Это необязательный параметр.

Пример:

```
[CONFIGURATION]  
CACHEMINTHRESHOLD=40
```

CACHESIZE

Объем дисковой памяти на сервере HTTP, предназначенной для временного хранения (*кэширования*) данных и документов ODWEK. Укажите значение в мегабайтах. Значение по умолчанию - 10 (десять мегабайт).

Примечание: Инструкции по разрешению кэширования документов приведены в разделе "CACHEDOCS" на стр. 14.

Область действия этого параметра является глобальной. В разделе CONFIGURATION его следует указывать только один раз.

Это необязательный параметр. Однако, чем больше дисковой памяти выделено, тем больше документов ODWEK может храниться на сервере HTTP. Как правило, такой подход позволяет увеличить скорость отправки уже просмотренных документов пользователям.

Пример:

```
[CONFIGURATION]  
CACHESIZE=1024
```

CACHEUSERIDS

Задает список ИД пользователей OnDemand, разделенный запятыми, для которых ODWEK применяет данные из кэш-памяти в процессе входа в систему. Для указанных ИД пользователей стандартная процедура входа в систему OnDemand пропускается, за исключением тех случаев, если данные отсутствуют в кэш-памяти или достигнуто значение Тайм-аут простоя (см. системные параметры сервера OnDemand). ИД пользователей разделяются с помощью запятых.

Примечания:

1. Если на сервере учитывается регистр символов ИД пользователя (см. системные параметры сервера OnDemand), их следует указывать точно в таком же виде, как он определен на сервере OnDemand.
2. ИД пользователей, перечисленные в списке CACHEUSERIDS, обладают правами доступа только к тем папкам, имена которых и другая информация хранятся в кэш-памяти. Пользователи не смогут получить доступ к папкам, созданным после входа на сервер OnDemand. Для того чтобы разрешить пользователям, перечисленным в списке CACHEUSERIDS, обращаться к новым папкам, удалите их имена из списка CACHEUSERIDS или очистите кэш.
3. Для применения данных из кэш-памяти для всех пользователей OnDemand укажите CACHEUSERIDS=*

Область действия этого параметра является глобальной. В разделе CONFIGURATION его следует указывать только один раз.

Это необязательный параметр.

Пример:

```
[CONFIGURATION]  
CACHEUSERIDS=user1,user2,user3
```

CODEPAGE

Задает кодовую страницу базы данных OnDemand. По умолчанию ODWEK применяет кодовую страницу сервера HTTP.

Область действия этого параметра является глобальной. В разделе CONFIGURATION его следует указывать только один раз. В случае применения API Вход в систему указанную кодовую страницу можно переопределить с помощью параметра `_codepage`.

Это необязательный параметр. Однако, если кодовая страница сервера HTTP отличается от кодовой страницы базы данных, параметр CODEPAGE необходимо указать.

Пример:

```
[CONFIGURATION]  
CODEPAGE=37
```

DOCSIZE

В процессе загрузки документов с сервера OnDemand определяет максимальный размер (в байтах) документа записываемого непосредственно в оперативную память, вместо записи на диск. Документы, размеры которых меньше указанного значения, загружаются непосредственно в оперативную память. В противном случае, документ

сначала записывается на диск, а затем для открытия в браузере считывается в оперативную память. Небольшое значение позволяет сохранить ресурсы системы. Более выковое значение увеличивает производительность функций просмотра документов. Укажите значение из диапазона от 0 до n байт, где n - это объем доступной оперативной памяти системы. Нулевое значение означает, что применяется размер по умолчанию 1 Мб. Если этот параметр не указан, либо не определено или не распознано значение, то применяется размер по умолчанию 1 Мб.

Область действия этого параметра является глобальной. В разделе CONFIGURATION его следует указывать только один раз.

Это необязательный параметр.

Пример:

```
[CONFIGURATION]
DOCSIZE=524287
```

IMAGEDIR

Определяет каталог, содержащий файлы изображений ODWEK.

Примечания:

1. ODWEK объединяет указанное значение с именами файлов изображений, обнаруженными в тегах изображений HTML. Например, если указать следующее значение:

```
imagedir=pictures
```

То тег изображений HTML для функции Просмотр документов в выводе будет выглядеть следующим образом:

```
<IMG SRC="pictures/odic_vd.gif">
```

2. Вы можете указать имя каталога или AliasMatch:

- Имя каталога следует указывать относительно каталога /QIBM/UserData/OnDemand/WWW. Например, если указан параметр imagedir=pictures, то изображения должны существовать в каталоге /QIBM/UserData/OnDemand/WWW/PICTURES.
- AliasMatch должен быть определен в файле конфигурации сервера HTTP. Например, если указан параметр imagedir=/pictures/, то в файле конфигурации сервера HTTP должен быть указан AliasMatch для /pictures/. В правиле AliasMatch должен быть указан полный путь к каталогу сервера. Например:

```
AliasMatch ^/images/(.*)$ /QIBM/UserData/OnDemand/WWW/PICTURES/$1
```

3. Проверьте права доступа к указанному каталогу. Процессы, запускающие программы ODWEK, должны иметь права доступа на чтение к каталогу изображений.

Область действия этого параметра является глобальной. В разделе CONFIGURATION его следует указывать только один раз.

Это обязательный параметр.

Пример:

```
[CONFIGURATION]
IMAGEDIR=pictures
```

LANGUAGE

Определяет язык для отображения сообщений ODWEK. По умолчанию применяется английский (ENU). ODWEK поддерживает следующие языки:

Значение	Регион
ARA	Египет
CHS	Китай
CHT	Тайвань
DAN	Дания
DEU	Германия
ENU	США/Английский
ESP	Испания
FIN	Финляндия
FRA	Франция
FRC	Канада
ITA	Италия
JPN	Япония
KOR	Корея
NLD	Нидерланды
NOR	Норвегия
PTB	Бразилия
SVE	Швеция

Область действия этого параметра является глобальной. В разделе CONFIGURATION его следует указывать только один раз.

Это необязательный параметр.

Пример:

```
[CONFIGURATION]  
LANGUAGE=JPN
```

ShowSearchString

Определяет, следует ли применять функцию Автоматический поиск. Функция Автоматический поиск поддерживает транзакции документов строковых данных и текстовый поиск в них с помощью программы просмотра строковых данных Java. Функция Автоматический поиск автоматически выделяет строки документа, соответствующие критериям поиска, указанным пользователем.

Если функция Автоматический поиск применяется и пользователь открывает документ из списка результатов для выполнения транзакции или поиска текста, то система автоматически находит текст документа, соответствующий указанным критериям поиска. Если совпадение найдено, то соответствующая строка выделяется. В противном случае выдается соответствующее сообщение. Если пользователь открывает другой документ для просмотра (или повторно открывает уже просмотренный документ), поиск выполняется снова.

Для того чтобы активировать функцию Автоматический поиск укажите в параметре ShowSearchString значение 1. Для отключения функции Автоматический поиск укажите в параметре ShowSearchString значение 0.

Область действия этого параметра является глобальной. В разделе CONFIGURATION его следует указывать только один раз.

Это необязательный параметр. Если он не указан, то по умолчанию применяется значение 0 (отключено).

Пример:

```
[CONFIGURATION]
ShowSearchString=1
```

TEMPDIR

Позволяет указать каталога для хранения временных файлов ODWEK.

Область действия этого параметра является глобальной. В разделе CONFIGURATION его следует указывать только один раз.

Это необязательный параметр. Если параметр TEMPDIR не указан, временные файлы ODWEK сохраняются в рабочем каталоге. В случае применения программы CGI рабочим каталогом является каталог, в котором установлена программа CGI. Для сервлета рабочим каталогом является каталог, в котором расположен этот сервлет. Точное расположение рабочего каталога зависит от сервера приложений Java: это может быть каталог, в котором расположен файл java.exe или каталог сервлетов.

Пример:

```
[CONFIGURATION]
TEMPDIR=/QIBM/UserData/OnDemand/WWW/TMP
```

Примечание: Проверьте права доступа к указанному каталогу. Процессы, запускающие программы ODWEK, должны иметь права доступа к временному каталогу на чтение и запись.

TEMPLATEDIR

Определяет каталог, содержащий файлы шаблонов HTML. ODWEK применяет файлы шаблонов для создания Web-страниц для различных функций продуктов (таких как Вход в систему, Поиск, Загрузить документ и т.д.) По умолчанию ODWEK загружает файлы шаблонов из каталога /QIBM/UserData/OnDemand/WWW/SAMPLES.

Примечание: Проверьте права доступа к указанному каталогу. Процессы, запускающие программы ODWEK, должны иметь права доступа на чтение к каталогу шаблонов.

Область действия этого параметра является глобальной. В разделе CONFIGURATION его следует указывать только один раз.

Это необязательный параметр.

Пример:

```
[CONFIGURATION]
TEMPLATEDIR=/QIBM/UserData/OnDemand/WWW/SAMPLES
```

[SECURITY]

В разделе SECURITY содержатся параметры защиты, применяемые ODWEK для работы с сервером HTTP.

Область действия этого раздела является глобальной. В файле ARSWWW.INI ее следует указывать только один раз.

Это необязательный раздел.

В данном разделе предусмотрены следующие параметры:

REPORTSERVERTIMEOUT

Позволяет указать, что ODWEK следует использовать параметр Тайм-аут простоя, определенный для сервера OnDemand. Параметр Тайм-аут простоя определяет момент завершения сеанса с неактивным пользователем. ODWEK применяет параметр Тайм-аут простоя, если в параметре REPORTSERVERTIMEOUT указано значение 1.

Область действия этого параметра является глобальной. В разделе SECURITY его следует указывать только один раз.

Это необязательный параметр. Если параметр REPORTSERVERTIMEOUT не указан, то параметр Тайм-аут простоя не применяется, т.е. ODWEK не прерывает сеансы с неактивными пользователями. Дополнительная информация о параметре Тайм-аут простоя приведена в электронной справке по клиенту администрирования.

Пример:

```
[SECURITY]  
REPORTSERVERTIMEOUT=1
```

SERVERACCESS

Задает список серверов OnDemand, разделенный запятыми, доступных для ODWEK. Если параметр SERVERACCESS указан, то клиенты, работающие с ODWEK, и программы, применяющие API, могут обращаться только к указанным серверам. Вы можете указать адрес TCP/IP, псевдоним или полное имя хоста сервера.

Область действия этого параметра является глобальной. В разделе SECURITY его следует указывать только один раз.

Это необязательный параметр.

Пример:

```
[SECURITY]  
SERVERACCESS=dave,gunnar
```

[AFP2HTML]

Раздел AFP2HTML содержит параметры, применяемые программой AFP2WEB Transform. Программа AFP2WEB Transform преобразует документы и ресурсы AFP в документы HTML, которые можно просмотреть с помощью аплета AFP2HTML.

Примечания:

1. Для преобразования документов AFP в документы HTML администратор должен получить служебную программу AFP2WEB Transform из IBM, установить и настроить ее на сервере. Более подробную информацию о служебной программе AFP2WEB Transform можно получить в представительстве IBM. Кроме того, для AFP2WEB Transform необходимо указать опции конфигурации. Дополнительная информация о файле конфигурации приведена в разделе Приложение E, “Преобразование AFP в HTML”, на стр. 129.
2. Для преобразования документов с помощью программы AFP2WEB Transform следует указать параметр AFPVIEWING=HTML в разделе DEFAULT BROWSER (или в разделах других браузеров). Более подробная информация приведена в разделе “AFPVIEWING” на стр. 31. (Если вы планируете работать с API Загрузить

документ, укажите параметр `_afr=HTML`. Дополнительная информация приведена в разделе “Загрузить документ” на стр. 77.)

3. AFP2HTML применяется по умолчанию для просмотра преобразованных документов. Если преобразованный документ сохранен на сервере OnDemand в качестве большого объекта, то апплет AFP2HTML добавляет удобные средства управления, позволяющие перемещаться по страницам документа.

Область действия этого раздела является глобальной. В файле ARSWWW.INI ее следует указывать только один раз.

Это необязательный раздел.

В данном разделе предусмотрены следующие параметры:

CONFIGFILE

Файл конфигурации, в котором указаны опции, применяемые программой AFP2WEB Transform для преобразования документов и ресурсов AFP в данные HTML, а также шрифты и изображения, которые можно просмотреть с помощью апплета AFP2HTML. На рисунке Приложение E, “Преобразование AFP в HTML”, на стр. 129 приведен пример файла конфигурации, поставляемого вместе с OnDemand. Подробное описание опций, которые можно указать в файле конфигурации, приведено в документации по AFP2WEB Transform.

Область действия этого параметра является глобальной. В разделе AFP2HTML его следует указывать только один раз.

Это необязательный параметр.

Пример:

```
[AFP2HTML]
CONFIGFILE=afp2html.ini
```

INSTALLDIR

Каталог, в котором расположены программы AFP2WEB Transform, файлы конфигурации и файлы преобразований. Укажите полный путь к каталогу сервера HTTP.

Примечание: Проверьте права доступа к указанному каталогу. Процессы, запускающие программы ODWEK, должны иметь права доступа на чтение к установочному каталогу.

Область действия этого параметра является глобальной. В разделе AFP2HTML его следует указывать только один раз.

Это необязательный параметр.

Пример:

```
[AFP2HTML]
INSTALLDIR=/QIBM/UserData/OnDemand/www/bin
```

USEEXECUTABLE

Определяет способ запуска программы AFP2WEB Transform: с помощью общей библиотеки (DLL) или выполняемого файла (EXE).

Важная информация: При работе с iSeries должен применяться выполняемый файл. Поэтому в этом параметре всегда должно быть указано значение 1.

Область действия этого параметра является глобальной. В разделе AFP2HTML его следует указывать только один раз.

Это необязательный параметр.

Пример:

```
[AFP2HTML]  
USEEXECUTABLE=1
```

[AFP2PDF]

Раздел AFP2PDF содержит параметры, применяемые программой IBM AFP2WEB Transform. Программа AFP2PDF Transform преобразует документы и ресурсы AFP в документы PDF, которые можно просмотреть с помощью апплета Просмотр Adobe Acrobat.

Примечания:

1. Для преобразования документов AFP в документы PDF администратор должен получить служебную программу AFP2PDF Transform из IBM, установить и настроить его на сервере. Более подробную информацию о служебной программе AFP2PDF Transform можно получить в представительстве IBM. Кроме того, для AFP2PDF Transform необходимо указать опции конфигурации. Дополнительная информация о файле конфигурации приведена в разделе Приложение F, “Преобразование AFP в PDF”, на стр. 133.
2. Для преобразования документов с помощью программы AFP2PDF Transform следует указать параметр AFPVIEWING=PDF в разделе DEFAULT BROWSER (или в разделах других браузеров). Более подробная информация приведена в разделе “AFPVIEWING” на стр. 31. (Если вы планируете работать с API Загрузить документ, укажите параметр _afr=PDF. Дополнительная информация приведена в разделе “Загрузить документ” на стр. 77.)
3. Для просмотра преобразованных документов по умолчанию применяется программа просмотра Adobe Acrobat. Получите программу просмотра для браузеров, применяемых в вашей организации.

Область действия этого раздела является глобальной. В файле ARSWWW.INI ее следует указывать только один раз.

Это необязательный раздел.

В данном разделе предусмотрены следующие параметры:

CONFIGFILE

Файл конфигурации, в котором указаны опции, применяемые программой AFP2PDF Transform для преобразования документов и ресурсов AFP в документы PDF, которые можно просмотреть с помощью апплета AFP2HTML. На рисунке Приложение F, “Преобразование AFP в PDF”, на стр. 133 приведен пример файла конфигурации, поставляемого вместе с OnDemand. Подробное описание опций, которые можно указать в файле конфигурации, приведено в документации по AFP2PDF Transform.

Область действия этого параметра является глобальной. В разделе AFP2PDF его следует указывать только один раз.

Это необязательный параметр.

Пример:

```
[AFP2PDF]
CONFIGFILE=afp2pdf.ini
```

INSTALLDIR

Каталог, в котором расположены программы AFP2PDF Transform, файлы конфигурации и файлы преобразований. Укажите полный путь к каталогу сервера HTTP.

Примечание: Проверьте права доступа к указанному каталогу. Процессы, запускающие программы ODWEK, должны иметь права доступа на чтение к установочному каталогу.

Область действия этого параметра является глобальной. В разделе AFP2PDF его следует указывать только один раз.

Это необязательный параметр.

Пример:

```
[AFP2PDF]
INSTALLDIR=/QIBM/UserData/OnDemand/www/bin
```

USEEXECUTABLE

Определяет способ запуска функции AFP2WEB: с помощью общей библиотеки (DLL) или выполняемого файла (EXE).

Важная информация: При работе с iSeries должен применяться выполняемый файл. Поэтому в этом параметре всегда должно быть указано значение 1.

Область действия этого параметра является глобальной. В разделе AFP2PDF его следует указывать только один раз.

Это необязательный параметр.

Пример:

```
[AFP2PDF]
USEEXECUTABLE=1
```

[MIMETYPES]

Раздел MIMETYPES позволяет определить тип данных Многоцелевых расширений почты Internet (MIME) для документов, загружаемых с сервера OnDemand. С помощью типа данных MIME браузер форматирует и отображает документ или выбирает соответствующую программу просмотра, апплет или пользовательскую программу для открытия документа.

Примечания:

1. В разделе MIMETYPES необходимо указать пару параметр=*значение* для каждого типа документов, загружаемых с сервера OnDemand. Параметр определяет тип данных документов OnDemand. (Это тип данных, присвоенный приложению OnDemand на странице Просмотр информации.) *Значение* определяет программу, с помощью которой открывается документ. В *значении* учитывается регистр символов.

2. В примере файла ARSWWW.INI (см. “Пример файла ARSWWW.INI” на стр. 38), в разделе MIMETYPES указаны параметры для всех стандартных типов данных, поддерживаемых сервером OnDemand (AFP, BMP, EMAIL, GIF, JFIF, LINE, PCX, PDF и TIFF).
3. Кроме стандартных типов данных поддерживаются также и пользовательские типы данных. Пользовательские типы данных позволяют определить любой тип данных для хранения в системе. Для того чтобы пользователи могли просматривать документы пользовательского типа в раздел MIMETYPE необходимо добавить соответствующий параметр. Параметр должен определять тип данных MIME, а также расширение файла, указанное для приложения OnDemand на странице Просмотр информации. Кроме того, расширение файла должно быть зарегистрировано в операционной системе клиента. Например, предположим, что определяется приложение для хранения на сервере OnDemand документов Lotus WordPro. На странице Просмотр информации указано расширение LWP. Для того чтобы система смогла распознать документы, загруженные приложением, в файл ARSWWW.INI следует добавить следующий параметр:

```
[MIMETYPES]
LWP=application/vnd.lotus-wordpro
```

При этом, если пользователь загружает из приложения такой документ, то ODWEK определяет тип данных MIME как application/vnd.lotus-wordpro и система открывает документ с помощью Lotus WordPro. Для Netscape тип данных MIME определяется с помощью Предпочтения->Навигатор->Приложения.

В разделе Табл. 1 перечислены типы данных MIME для нескольких приложений:

Таблица 1. Типы данных MIME для нескольких приложений

Приложение	Тип данных MIME
Приложения Lotus	WK1=application/vnd.lotus-1-2-3 WK3=application/vnd.lotus-1-2-3 WK4=application/vnd.lotus-1-2-3 123=application/vnd.lotus-1-2-3 APR=application/vnd.lotus-approach VEW=application/vnd.lotus-approach LWP=application/vnd.lotus-wordpro SAM=application/vnd.lotus-wordpro MWP=application/vnd.lotus-wordpro SMM=application/vnd.lotus-wordpro PRE=application/vnd.lotus-freelance PRZ=application/vnd.lotus-freelance
Приложения Microsoft	DOC=application/msword XLS=application/vnd.ms-excel PPS=application/vnd.ms-powerpoint PPT=application/vnd.ms-powerpoint MPD=application/vnd.ms-project MPP=application/vnd.ms-project MPT=application/vnd.ms-project MPD=application/vnd.ms-project
Приложения HTML	HTML=application/html HTM=application/htm

Область действия этого раздела является глобальной. В файле ARSWWW.INI ее следует указывать только один раз.

Это необязательный раздел.

В данном разделе предусмотрены следующие параметры:

AFP

Тип данных MIME для документов AFP, если в разделе [DEFAULT BROWSER] указан параметр AFPVIEWING=NATIVE. За дополнительной информацией обратитесь к разделу “AFPVIEWING” на стр. 31. Задает тип данных MIME документа, с помощью которого браузер определяет программу для обработки документа.

Область действия этого параметра является глобальной. В разделе MIMETYPES его следует указывать только один раз.

Это необязательный параметр.

Пример:

```
[MIMETYPES]
AFP=application/afp
```

BMP

Тип данных MIME для документов BMP. По умолчанию документы BMP отображаются с помощью программы Image Web Viewer.

Область действия этого параметра является глобальной. В разделе MIMETYPES его следует указывать только один раз.

Это необязательный параметр. Однако, если этот параметр не указан, то ODWEK задает тип данных image/bmp и запускает программу, указанную для файлов BMP в операционной системе клиента.

Пример:

```
[MIMETYPES]
BMP=image/IBM-OnDemand
```

GIF

Тип данных MIME для документов GIF. По умолчанию документы GIF отображаются с помощью программы Image Web Viewer.

Область действия этого параметра является глобальной. В разделе MIMETYPES его следует указывать только один раз.

Это необязательный параметр. Однако, если этот параметр не указан, то ODWEK задает тип данных image/gif и отображает документы GIF с помощью встроенной программы просмотра браузера.

Пример:

```
[MIMETYPES]
GIF=image/IBM-OnDemand
```

EMAIL

Тип данных MIME для документов EMAIL. Дополнительная информация об обработке документов EMAIL перед отправкой их клиентам приведена в разделе “EMAILVIEWING” на стр. 32.

Примечания:

1. В случае преобразования документов EMAIL в формат HTML определяется тип данных text/html. При этом значение параметра EMAIL игнорируется.
2. В случае извлечения несжатых документов EMAIL значение параметра EMAIL позволяет определить программу для их открытия.

Область действия этого параметра является глобальной. В разделе MIMETYPES его следует указывать только один раз.

Это необязательный параметр.

Пример:

```
[MIMETYPES]  
EMAIL=text/plain
```

JFIF

Тип данных MIME для документов JFIF (JPEG). По умолчанию документы JFIF отображаются с помощью программы Image Web Viewer.

Область действия этого параметра является глобальной. В разделе MIMETYPES его следует указывать только один раз.

Это необязательный параметр. Однако, если этот параметр не указан, то ODWEK задает тип данных image/jpeg и запускает программу, указанную для файлов JPEG в операционной системе клиента.

Пример:

```
[MIMETYPES]  
JFIF=image/IBM-OnDemand
```

LINE

Тип данных MIME для документов строковых данных. Дополнительная информация об обработке документов строковых данных перед отправкой их клиентам приведена в разделе "LINEVIEWING" на стр. 34.

Данный параметр применяется, если в разделе [DEFAULT BROWSER] указан параметр LINEVIEWING=NATIVE. В случае извлечения несжатых документов строковых данных значение параметра LINE позволяет определить программу для их открытия.

Область действия этого параметра является глобальной. В разделе MIMETYPES его следует указывать только один раз.

Это необязательный параметр.

Пример:

```
[MIMETYPES]  
LINE=text/html
```

PCX

Тип данных MIME для документов PCX. По умолчанию документы PCX отображаются с помощью программы Image Web Viewer.

Область действия этого параметра является глобальной. В разделе MIMETYPES его следует указывать только один раз.

Это необязательный параметр. Однако, если этот параметр не указан, то ODWEK задает тип данных `image/pcx` и запускает программу, указанную для файлов PCX в операционной системе клиента.

Пример:

```
[MIMETYPES]
PCX=image/IBM-OnDemand
```

PDF

Тип данных MIME для документов PDF.

Примечания:

1. Значение параметра PDF позволяет определить программу для открытия документов PDF. По умолчанию документы открываются с помощью программы просмотра Adobe Acrobat.
2. Для просмотра документов PDF необходимо получить и установить программу просмотра Adobe Acrobat для браузеров, применяемых в вашей организации.

Область действия этого параметра является глобальной. В разделе MIMETYPES его следует указывать только один раз.

Это необязательный параметр.

Пример:

```
[MIMETYPES]
PDF=application/pdf
```

TIFF

Тип данных MIME для документов TIFF. По умолчанию документы TIFF отображаются с помощью программы Image Web Viewer.

Область действия этого параметра является глобальной. В разделе MIMETYPES его следует указывать только один раз.

Это необязательный параметр. Однако, если этот параметр не указан, то ODWEK задает тип данных `image/tiff` и запускает программу, указанную для файлов TIFF в операционной системе клиента.

Пример:

```
[MIMETYPES]
TIFF=image/IBM-OnDemand
```

[ATTACHMENT IMAGES]

Раздел ATTACHMENT IMAGES позволяет указать файлы изображений, применяемые для отображения вложений в документы ODWEK. Каждый файл изображения должен содержать значок, представляющий конкретный тип вложения. Например, вы можете указать файл изображения, содержащий значок для текстового вложения, графического вложения и т.д.

Примечания:

1. Каждый указанный параметр должен определять тип файлов, который операционная система связывает с данным типом вложений. Тип файлов определяет программу, запускаемую операционной системой для обработки вложений. Например, если операционная система связывает файлы TXT с текстовыми вложениями, добавьте в раздел ATTACHMENT IMAGES параметр `TXT=значение`. В качестве значения укажите имя файла, содержащего значок,

обозначающий текстовое вложение в документ. При нажатии на этот значок операционная система запускает программу, зарегистрированную для открытия документов TXT.

2. По умолчанию все вложения обозначаются файлом `odic_att.gif` (который расположен в каталоге, указанном с помощью параметра `IMAGEDIR` в разделе `CONFIGURATION`). Кроме того, для типов файлов, не указанных в разделе `ATTACHMENT IMAGES`, применяется файл `odic_att.gif`.

Область действия этого раздела является глобальной. В файле `ARSWWW.INI` ее следует указывать только один раз.

Это необязательный раздел.

В данном разделе предусмотрены следующие параметры:

BMP

Этот параметр задает тип данных BMP. Данное значение определяет файл, в котором содержится значок, обозначающий вложенное в документ изображение BMP.

Область действия этого параметра является глобальной. В разделе `ATTACHMENT IMAGES` его следует указывать только один раз.

Это необязательный параметр.

Пример:

```
[ATTACHMENT IMAGES]
BMP=userBitMap.gif
```

GIF

Этот параметр задает тип данных GIF. Данное значение определяет файл, в котором содержится значок, обозначающий вложенное в документ изображение GIF.

Область действия этого параметра является глобальной. В разделе `ATTACHMENT IMAGES` его следует указывать только один раз.

Это необязательный параметр.

Пример:

```
[ATTACHMENT IMAGES]
GIF=userGIF.gif
```

TXT

Этот параметр задает тип данных TXT. Данное значение определяет файл, в котором содержится значок, обозначающий вложенный в документ текстовый файл.

Область действия этого параметра является глобальной. В разделе `ATTACHMENT IMAGES` его следует указывать только один раз.

Это необязательный параметр.

Пример:

```
[ATTACHMENT IMAGES]
TXT=userText.gif
```

[NO HTML]

В разделе NO HTML содержатся параметры, позволяющие переопределить символы, разграничивающие строки и разделяющие список значений в выводе ASCII с ограничителями. Вывод ASCII с ограничителями создается, если в параметре `_nohtml` указано значение 1. Дополнительная информация о выводе ASCII с ограничителями в разделе Приложение Н, “Не создавать вывод HTML”, на стр. 137.

Область действия этого раздела является глобальной. В файле ARSWWW.INI ее следует указывать только один раз.

Это необязательный раздел.

В данном разделе предусмотрены следующие параметры:

BEGIN

Задаёт символ, применяемый ODWEK для ограничения начала строки или строки значений. Ограничитель BEGIN следует изменить, если строка содержит символ по умолчанию (символ `[`).

Область действия этого параметра является глобальной. В разделе NO HTML его следует указывать только один раз.

Это необязательный параметр.

Пример:

```
[NO HTML]
BEGIN=<
```

END

Задаёт символ, применяемый ODWEK для ограничения конца строки или строки значений. Ограничитель END следует изменить, если строка содержит символ по умолчанию (символ `]`).

Область действия этого параметра является глобальной. В разделе NO HTML его следует указывать только один раз.

Это необязательный параметр.

Пример:

```
[NO HTML]
END=>
```

SEPARATOR

Задаёт символ, применяемый ODWEK для разделения строки значений. Ограничитель SEPARATOR следует изменить, если строка содержит символ по умолчанию (символ `^`).

Область действия этого параметра является глобальной. В разделе NO HTML его следует указывать только один раз.

Это необязательный параметр.

Пример:

```
[NO HTML]
SEPARATOR=;
```

[DEFAULT BROWSER]

Раздел DEFAULT BROWSER позволяет указать параметры браузеров, применяемых в вашей организации. Указанные параметры применяются только в том случае, если они не указаны в разделе конкретного браузера. Более подробная информация приведена в разделе “[browser]” на стр. 36. (Параметры, указанные в разделах браузеров, переопределяют соответствующие параметры раздела DEFAULT BROWSER.)

Область действия этого раздела распространяется на все браузеры. В файле ARSWWW.INI ее следует указывать только один раз.

Это необязательный раздел.

В данном разделе предусмотрены следующие параметры:

ADDEXTENSION

Определяет, следует ли добавлять в дополнительную информацию о пути, возвращаемую браузеру, трехсимвольное расширение пути. Добавление расширения файла в URL позволяет определить для документа необходимую программу просмотра. Значение по умолчанию - 0 (ноль). Оно означает, что расширение файла в URL не добавляется.

Примечание: При работе с браузером Microsoft Internet Explorer IBM рекомендует указывать параметр ADDEXTENSION=1 для добавления URL в расширение файла.

Область действия этого параметра является глобальной. В разделе DEFAULT BROWSER его следует указывать только один раз.

Это необязательный параметр.

Пример:

```
[DEFAULT BROWSER]
ADDEXTENSION=1
```

ADDFIELDSTODOCID

Определяет, следует ли в идентификаторы документов добавлять значения полей. (Идентификаторы документов возвращаются функцией Список найденных документов.) Значение по умолчанию - 0 (ноль). Оно означает, что значения полей в идентификаторы документов не добавляются. Значения полей кроме идентификаторов документов добавляются в протокол системы, если в настройках системы указано, что сообщения группы приложений должны заноситься в протокол системы.

Примечания:

1. В случае применения функции Обновить документ следует указать ADDFIELDSTODOCID=1.
2. Если для опции Флаги комментариев в поле таблицы базы данных документа указано значение Да, **необходимо** указать ADDFIELDSTODOCID=1. Опцию Флаги комментариев в поле таблицы базы данных документа можно выбрать в окне диалога Информация о базе данных на странице Общие в определениях группы приложений OnDemand. (Для открытия окна диалога Информация о базе данных нажмите кнопку Дополнительно.)

Область действия этого параметра является глобальной. В разделе DEFAULT BROWSER его следует указывать только один раз.

Это необязательный параметр.

Пример:

```
[DEFAULT BROWSER]
ADDFIELDSTODOCID=1
```

ADDNOTES

Определяет, следует ли добавлять в документы комментарии. Добавлением аннотаций в документы из списка документов управляет программное обеспечение ODWEK. Значение по умолчанию - 0 (ноль). Оно означает, что комментарии запрещено добавлять в документы.

Примечание: Права доступа на добавление комментариев определены в группе приложений OnDemand.

Область действия этого параметра является глобальной. В разделе DEFAULT BROWSER его следует указывать только один раз.

Это необязательный параметр.

Пример:

```
[DEFAULT BROWSER]
ADDNOTES=1
```

AFPVIEWING

Данное значение определяет действие, выполняемое в процессе загрузки документа AFP с сервера OnDemand, перед отправкой документа клиенту. Например, можно преобразовать документы AFP в формат HTML с помощью служебной программы AFP2WEB Transform и отобразить вывод HTML с помощью апплета AFP2HTML. Для этого следует указать значение AFPVIEWING=HTML, чтобы перед отправкой клиенту документы AFP преобразовывались.

Для этого параметра предусмотрены следующие значения:

- | | |
|---------------|---|
| ASCII | Документы AFP преобразуются в текст ASCII. |
| HTML | Документы AFP преобразуются в документы HTML с помощью программы AFP2WEB Transform. |
| NATIVE | Документы и ресурсы AFP загружаются с сервера OnDemand и распаковываются. |

Примечание: Если указан параметр AFPVIEWING=NATIVE, убедитесь, что для типа данных MIME документов AFP определена необходимая программа просмотра. Более подробная информация приведена в разделе “[MIMETYPES]” на стр. 23.

- | | |
|------------|--|
| PDF | Документы AFP преобразуются в документы PDF с помощью программы AFP2WEB Transform. |
|------------|--|

Примечание: Если указан параметр AFPVIEWING=PDF, убедитесь, что для типа данных MIME документов PDF определена необходимая программа просмотра. Более подробная информация приведена в разделе “[MIMETYPES]” на стр. 23.

- | | |
|---------------|--|
| PLUGIN | Документы AFP не преобразуются (по умолчанию). |
|---------------|--|

Область действия этого параметра является глобальной. В разделе DEFAULT BROWSER его следует указывать только один раз. В случае применения функции Обновить документ вы можете переопределить указанное действие с помощью параметра `_afr`.

Это необязательный параметр.

Пример:

```
[DEFAULT BROWSER]
AFPVIEWING=PLUGIN
```

AUTODOCRETRIEVAL

Позволяет автоматически отображать документ, если он является единственным совпадением, полученным в результате запроса. Такой подход позволяет для запросов, в результате которых заведомо будет возвращен только один документ, пропустить Web-страницу списка документов и отобразить документ без дополнительных действий пользователя. Значение по умолчанию - 0 (ноль). Оно означает, что Web-страница со списком документов отображается даже в том случае, если запросу соответствует только один документ.

Важная информация: Если вы планируете работать с браузером Microsoft Internet Explorer разрешать применение автоматической загрузки документа не следует. IBM предполагает, что во всех разделах, определенных для Internet Explorer, будет указано значение AUTODOCRETRIEVAL=0.

Область действия этого параметра является глобальной. В разделе DEFAULT BROWSER его следует указывать только один раз.

Это необязательный параметр.

Пример:

```
[DEFAULT BROWSER]
AUTODOCRETRIEVAL=1
```

EMAILVIEWING

Данное значение определяет действие, выполняемое в процессе загрузки документа EMAIL с сервера OnDemand, перед отправкой документа клиенту.

Для этого параметра предусмотрены следующие значения:

NATIVE Документы EMAIL загружаются с сервера OnDemand и распаковываются.

Примечание: Если указан параметр EMAIL=NATIVE, убедитесь, что для типа данных MIME определена необходимая программа просмотра. Более подробная информация приведена в разделе “[MIMETYPES]” на стр. 23.

HTML Документы EMAIL преобразуются в формат HTML. Это значение по умолчанию.

Область действия этого параметра является глобальной. В разделе DEFAULT BROWSER его следует указывать только один раз. В случае применения функции Обновить документ вы можете переопределить указанное действие с помощью параметра `_email`.

Это необязательный параметр.

Пример:

```
[DEFAULT BROWSER]
EMAILVIEWING=HTML
```

ENCRYPTCOOKIES

Позволяет шифровать cookies, отправляемые браузеру. Значение по умолчанию - 0 (ноль). Оно означает, что cookie не шифруются. Для шифрования cookie, отправляемых браузеру, укажите значение 1.

Область действия этого параметра является глобальной. В разделе DEFAULT BROWSER его следует указывать только один раз.

Это необязательный параметр.

Пример:

```
[DEFAULT BROWSER]
ENCRYPTCOOKIES=1
```

ENCRYPTURL

Позволяет шифровать значения сервера, ИД пользователя, пароля и ИД документа, содержащиеся в URL, отправляемых браузеру. Значение по умолчанию - 0 (ноль). Оно означает, что данные значения не шифруются. Для их шифрования укажите значение 1.

Область действия этого параметра является глобальной. В разделе DEFAULT BROWSER его следует указывать только один раз.

Это необязательный параметр. Однако, если необходимо передавать параметры форм и значения между браузером и сервером HTTP с помощью метода GET, для их шифрования вы можете указать параметр ENCRYPTURL=1. Дополнительная информация об атрибуте метода тега формы приведена в разделе “Защита сервера и данных” на стр. 7.

Пример:

```
[DEFAULT BROWSER]
ENCRYPTURL=1
```

FOLDERDESC

Позволяет указать, следует ли отображать описание справа от имени папки на странице выбора папки. Значение по умолчанию - 0 (ноль). Оно означает, что описание папки не отображается. Для отображения описания укажите значение 1. Если этот параметр не указан, либо не определено или не распознано значение, то описание папки не показывается.

Область этого параметра является глобальной, если он не переопределен в разделе браузера (см. “[browser]” на стр. 36). В разделе DEFAULT BROWSER его следует указывать только один раз.

Это необязательный параметр.

Пример:

```
[DEFAULT BROWSER]
FOLDERDESC=1
```

LINEVIEWING

Данное значение определяет действие, выполняемое в процессе загрузки документа строковых данных с сервера OnDemand, перед отправкой его клиенту.

Для этого параметра предусмотрены следующие значения:

APPLET	Данные строковых данных преобразуются для просмотра с помощью апплета Строковые данные (по умолчанию).
ASCII	Документы строковых данных преобразуются в формат ASCII.
NATIVE	Документы строковых данных загружаются с сервера OnDemand и распаковываются.

Примечание: Если указан параметр LINEVIEWING=NATIVE, убедитесь, что для типа данных MIME определена необходимая программа просмотра. Более подробная информация приведена в разделе “[MIMETYPES]” на стр. 23.

Область действия этого параметра является глобальной. В разделе DEFAULT BROWSER его следует указывать только один раз. В случае применения функции Обновить документ вы можете переопределить указанное действие с помощью параметра `_line`.

Это необязательный параметр.

Пример:

```
[DEFAULT BROWSER]
LINEVIEWING=APPLET
```

MAXHITS

Задет максимальное число элементов, отображаемых в списке документов. Данное значение не зависит от числа совпадений.

Примечания:

1. В список документов добавляются элементы, соответствующие критериям запроса, в порядке их загрузки в базу данных.
2. С помощью первого указанного значения определяется число элементов, возвращаемых в списке документов:
 - a. Для функции Список найденных документов значение поля Максимальное число совпадений (указанное на странице Права доступа к папке). Это значение переопределяет все остальные значения.
 - b. Для функции Список найденных документов и Печать документа значение параметра `_max_hits`, если оно указано. Значение параметра `_max_hits` переопределяет параметр MAXHITS.
 - c. Значение параметра MAXHITS, если оно указано.
 - d. Если не указано ни одно из приведенных выше значений, то по умолчанию в список состоит из 200 элементов.

Область действия этого параметра является глобальной. В разделе DEFAULT BROWSER его следует указывать только один раз.

Это необязательный параметр.

Пример:

```
[DEFAULT_BROWSER]
MAXHITS=200
```

NOLINKS

Позволяет добавить в список документов средства для их просмотра. При этом напротив каждого документа отображается соответствующее средство, с помощью которого вы можете просмотреть документ. Значение по умолчанию - 0 (ноль). Оно означает, что для просмотра документа пользователь должен воспользоваться текстовой ссылкой.

Важная информация: При работе с Microsoft Internet Explorer следует указать значение NOLINKS=0. IBM предполагает, что во всех разделах, определенных для Internet Explorer, будет указано значение NOLINKS=0.

Область действия этого параметра является глобальной. В разделе DEFAULT BROWSER его следует указывать только один раз.

Это необязательный параметр.

Пример:

```
[DEFAULT_BROWSER]
NOLINKS=1
```

ODApplet.jre.path.IE

См. “Программа просмотра строковых данных Java” на стр. 52.

ODApplet.jre.path.NN

См. “Программа просмотра строковых данных Java” на стр. 52.

ODApplet.jre.version

См. “Программа просмотра строковых данных Java” на стр. 52.

ODApplet.version

См. “Программа просмотра строковых данных Java” на стр. 52.

SERVERPRINT

Позволяет добавить в список документов средства для их печати на сервере. При этом напротив каждого документа отображается соответствующее средство. Значение по умолчанию - 0 (ноль). Оно означает, что перед отправкой документа на принтер сервера пользователь должен открыть его.

Примечания:

1. Для этого на сервере OnDemand должен быть определен принтер сервера.
2. Права доступа на печать документов определены в группе приложений OnDemand.

Область действия этого параметра является глобальной. В разделе DEFAULT BROWSER его следует указывать только один раз.

Это необязательный параметр.

Пример:

```
[DEFAULT_BROWSER]
SERVERPRINT=1
```

SERVERPRINTERS

Позволяет указать тип устройств принтера сервера, доступных пользователям. Предусмотрены следующие три устройства принтера сервера:

- P** Принтер сервера
- I** Принтер сервера с информацией
- F** Факс сервера

Вы можете указать до трех типов в списке, разделенном запятыми.

В следующем примере показано, каким образом можно сделать доступными для пользователя принтер сервера или факс сервера:

```
[DEFAULT BROWSER]
SERVERPRINTERS=P,F
```

SHOWDOCLOCATION

Позволяет добавить в вывод расположение сохранения документа при создании вывода ASCII с ограничителями вместо HTML (см. Приложение Н, “Не создавать вывод HTML”, на стр. 137). Более подробная информация приведена в разделе “Список найденных документов” на стр. 139. Значение по умолчанию - 0 (ноль). Оно означает, что расположение не добавляется в вывод.

Примечание: Для того чтобы просмотреть расположение сохранения необходимо задать свойство Показать расположение документа в папке OnDemand.

Область действия этого параметра является глобальной. В разделе DEFAULT BROWSER его следует указывать только один раз.

Это необязательный параметр.

Пример:

```
[DEFAULT BROWSER]
SHOWDOCLOCATION=1
```

VIEWNOTES

Определяет, разрешено ли просматривать комментарии документов. Просмотром комментариев документов из списка документов управляет программное обеспечение ODWEK. Значение по умолчанию - 0 (ноль). Оно означает, что комментарии просматривать запрещено.

Примечание: Права доступа на просмотр комментариев определены в группе приложений OnDemand.

Область действия этого параметра является глобальной. В разделе DEFAULT BROWSER его следует указывать только один раз.

Это необязательный параметр.

Пример:

```
[DEFAULT BROWSER]
VIEWNOTES=1
```

[browser]

Позволяет указать опции для браузеров, применяемых в вашей организации. Параметры, указанные в разделе browser, переопределяют параметры раздела

DEFAULT BROWSER из файла ARSWWW.INI. (Параметры и значения, указанные в разделе DEFAULT BROWSER, применяются только в том случае, если они не указаны в разделе browser.)

Примечания:

1. Заголовок раздела должен содержать строку, определяющую браузер, для которого указываются опции. Для определения применяемого браузера ODWEK использует значение переменной среды HTTP_USER_AGENT. Затем в файле ARSWWW.INI выполняется поиск соответствующего раздела. Если раздел browser не найден, то в файле ARSWWW.INI выполняется поиск одного из следующих разделов:

[browser version(major.minor)/platform]

[browser version(major.minor)]

[browser version(major)]

[browser]

[DEFAULT BROWSER]

Применяются опции из первого раздела, соответствующего значению.

2. В качестве браузера можно указать IE или Netscape.
3. В качестве платформы можно указать WinNT или Unix.

Область действия раздела browser является глобальной для указанного браузера. Для каждого браузера следует указывать только один раздел browser. Рекомендуется указывать только те параметры, которые необходимо переопределить в разделе DEFAULT BROWSER.

Это необязательный раздел.

В этом разделе могут быть указаны те же параметры, что и в разделе браузера по умолчанию. См. “[DEFAULT BROWSER]” на стр. 30.

Примеры:

```
[IE 5]
AUTODOCRETRIEVAL=0
NOLINKS=0
```

```
[Netscape 4.7]
AUTODOCRETRIEVAL=1
NOLINKS=1
```

[DEBUG]

Раздел DEBUG содержит опции, предназначенные для устранения неполадок, возникших у вас и других пользователей при работе с ODWEK. Раздел DEBUG должен быть первым исполняемым оператором файла arswwww.ini.

Область действия этого раздела является глобальной. В файле ARSWWW.INI ее следует указывать только один раз.

Это необязательный раздел.

В данном разделе предусмотрены следующие параметры:

LOG

Позволяет записывать в файл протокола сообщения и другую информацию о программах. (Файл протокола называется ARSWWW.LOG.)

Область действия этого параметра является глобальной. В разделе DEBUG его следует указывать только один раз.

Это необязательный параметр. По умолчанию сообщения в файл протокола не записываются. Для того чтобы заносить сообщения в протокол укажите значение 1.

LOGDIR

Определяет каталог, в котором расположен файл ARSWWW.LOG, если ведение протокола разрешено с помощью параметра LOG.

Область действия этого параметра является глобальной. В разделе DEBUG его следует указывать только один раз.

Это необязательный параметр. По умолчанию, если ведение протокола разрешено, файл протокола расположен в каталоге /QIBM/UserData/OnDemand/WWW/LOG.

Пример:

```
[DEBUG]
LOGDIR=/QIBM/UserData/OnDemand/WWW/LOG
LOG=1
```

Пример файла ARSWWW.INI

Пример файла конфигурации ARSWWW.INI поставляется вместе с продуктом. В примере файла конфигурации указаны наиболее часто встречающиеся значения для серверов, Web-браузеров и программ просмотра.

```
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;; Конфигурация сервера   ;;;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

;[DEBUG]
;log=1
;logdir=/QIBM/UserData/OnDemand/www/logs

[@SRV@_<host alias>];
HOST=<host name>;
PORT= ;
PROTOCOL= ;

[@SRV@_myiSeries.mycompany.com]
HOST=myiSeries.mycompany.com
PORT=1450
PROTOCOL=0

[CONFIGURATION]
CodePage=37
Language=ENU
TemplateDir=/QIBM/UserData/OnDemand/www/SAMPLES
ImageDir=/IMAGES/
AppletDir=/applets/
CacheDir=/QIBM/UserData/OnDemand/www/
CacheSize=0
CacheMinThreshold=0
CacheMaxThreshold=0
CacheDocs=0
CacheUserIDs=web,demo,mstephens
```

```

[SECURITY]
SERVERACCESS=

[AFP2HTML]
InstallDir=/QIBM/UserData/OnDemand/www/bin
ConfigFile=/QIBM/UserData/OnDemand/www/bin/afp2html.ini
UseExecutable=1

[AFP2PDF]
InstallDir=/QIBM/UserData/OnDemand/www/bin
ConfigFile=/QIBM/UserData/OnDemand/www/bin/afp2pdf.ini
UseExecutable=1

[MIMETYPES]
BMP=image/IBM_OnDemand
GIF=image/IBM_OnDemand
JFIF=image/IBM_OnDemand
PCX=image/IBM_OnDemand
TIFF=image/IBM_OnDemand
PNG=image/IBM_OnDemand
PDF=application/pdf
AFP=application/afp
LINE=application/line
EMAIL=text/html
META=application/unknown

[ATTACHMENT IMAGES]
TXT=userText.gif
BMP=userBitMap.gif
GIF=userGIF.gif

[NO HTML]
BEGIN=[
END=]
SEPARATOR=^

;;;;;;;;;;;;;;;;;;;;;;;;;
;;; Браузер по умолчанию ;;;
;;;;;;;;;;;;;;;;;;;;;;;;;

[DEFAULT BROWSER]
;AfpViewing=[ascii,html,native,pdf,plugin,xenos]
AfpViewing=plugin
;LineViewing=[ascii,applet,native]
LineViewing=applet
;EmailViewing=[html,native]
EmailViewing=html
;MetaViewing=[xenos,native]
MetaViewing=native
NoLinks=1
ViewNotes=1
AddNotes=1
ServerPrint=1
ServerPrinters=P
AutoDocRetrieval=1
MaxHits=200
ShowDocLocation=1

[IE]
NoLinks=0
AddExtension=1
AddFieldsToDocid=0

```

Следующий шаг

После установки программного обеспечения ODWEK и настройки файла ARSWWW.INI следует настроить примеры приложений. См. Глава 3, “Настройка примеров приложений”, на стр. 41.

Глава 3. Настройка примеров приложений

В этом разделе приведены инструкции по настройке примеров приложений, поставляемых вместе с программным обеспечением ODWEK:

- LOGON.HTM. Это приложение предназначено для пользователей, которым разрешен доступ к нескольким папкам. Каждый пользователь определяется на сервере OnDemand. После входа на сервер отображается список доступных папок.
- CREDIT.HTM. Это приложение предназначено для нерегулярной работы с OnDemand. Пользователю отображаются критерии поиска, заданные для конкретной папки. В приложении кодируются следующие параметры: имя сервера OnDemand, ИД пользователя и пароль, имя папки и поля папки. В разделе "CREDIT.HTM" приведены инструкции по настройке этого приложения.

После внесения изменений в примеры приложений, опубликуйте URL соответствующих файлов, чтобы пользователи могли получить доступ к серверу OnDemand. Уровень настройки отличается для каждого приложения. Инструкции по настройке предусмотрены только для одного приложения. При необходимости с помощью этих инструкций вы можете настроить и другие приложения.

Примечание: Кроме внесения изменений в примеры приложений IBM рекомендует изменить файл TEMPLATE.HTM, используемый в вашей организации. Файл TEMPLATE.HTM содержит пользовательскую информацию, применяемую ODWEK для отображения Web-страниц. Важная информация, связанная с изменением этого файла приведена в разделе "TEMPLATE.HTM" на стр. 42.

LOGON.HTM

1. Скопируйте файл logon.htm из установочного каталога (/QIBM/ProdData/OnDemand/www.samples) в корневой каталог документов сервера HTTP.
2. Для программы CGI убедитесь, что в файле logon.htm содержатся следующие строки:

```
<h4>Введите информацию для входа в систему:</h4>
<FORM METHOD=POST ACTION="/arswww.cgi">
```
3. Для сервлета убедитесь, что в файле logon.htm содержатся следующие строки:

```
<FORM METHOD=POST ACTION="/ArsWWWServlet">
```

CREDIT.HTM

Для настройки примера приложения CREDIT.HTM необходимо скопировать его во все папки, которые должны быть доступны для пользователей. Имя файла должно совпадать с именем папки.

1. Измените файл CREDIT.HTM. (по умолчанию он расположен в каталоге /QIBM/UserData/OnDemand/WWW/SAMPLES.)
2. Измените или удалите фоновое изображение, указанное в операторе <body> (строка 11).
3. При необходимости измените цвет фона, указанный в операторе <body> (строка 11).
4. Измените или удалите изображение продукта, указанное в операторе (строка 12).

5. Замените имя папки, указанное в операторе <h1> (строка 15).
6. Замените текст, указанный в операторах <p> (строки 17-25). Введите инструкции для пользователя.
7. Замените имя каталога CGI-BIN, указанное в операторе <FORM> (строка 29). Введите имя каталога CGI-BIN на сервере HTTP, в котором содержатся программы и файлы ODWEK.
8. Замените значение, указанное в операторе <input> (строка 30). Это строка с именами отображаемых полей папки, разделенных запятыми.
9. Замените значение, указанное в операторе <input> (строка 31). Это имя папки.
10. Замените значение, указанное в операторе <input> (строка 33). Это максимальное число элементов, отображаемых в списке документов. Данное значение не зависит от числа совпадений.
11. Замените имя сервера, указанное в операторе <input> (строка 35). Это имя сервера OnDemand, с которым программное обеспечение ODWEK будет обмениваться данными. Имя сервера по умолчанию - gunnar.
12. Для сортировки элементов списка документов проверьте значение, указанное в операторе <input> (строка 36). В противном случае, удалите строку 36.
13. Для сортировки элементов списка документов проверьте значение, указанное в операторе <input> (строка 37). В противном случае, удалите строку 37.
14. Замените значение, указанное в операторе <input> (строка 38). Это ИД пользователя OnDemand. Указанный ИД пользователя должен обладать правами доступа к папке и данным группы приложений.
15. При необходимости измените имя файла шаблона, указанное в операторе <input> (строка 39). Файл шаблона необходим OnDemand для создания последовательности Web-страниц. По умолчанию указан файл template.htm.
16. Измените строки 40-43, в которых указаны поля поиска первой папки.
 - a. Укажите имя поля папки в операторе .
 - b. Замените значение, указанное в поле Имя оператора <input> на имя поля папки.
 - c. Замените значение, указанное в поле Значение оператора <input> на значение поиска по умолчанию.
17. Скопируйте строки 40-43 и повторите действия, описанные для шага 16, для создания дополнительных полей поиска.
18. Сохраните внесенные изменения и закройте текстовый редактор.

TEMPLATE.HTM

Файл TEMPLATE.HTM представляет собой файл шаблона по умолчанию, применяемый ODWEK для создания Web-страниц, соответствующих различным функциям продукта (например, Вход в систему). Замените этот файл на файл, в котором указана пользовательская информация. Однако файл шаблона должен содержать следующую строку комментария HTML:

```
<!-- - - A0I# Marker - - -->
```

Расположение строки комментария определяет расположение вывода программы ODWEK. Строки, указанные до строки комментария, отображаются до вывода ODWEK. Строки, указанные после строки комментария, отображаются после вывода ODWEK.

По умолчанию файл шаблона расположен в каталоге, заданном с помощью параметра TEMPLATEDIR в файле ARSWWW.INI file. Дополнительная информация приведена в разделе "TEMPLATEDIR" на стр. 19.

Следующий шаг

После установки программного обеспечения ODWEK, настройки файла ARSWWW.INI и настройки примеров приложений установите программы просмотра Web на рабочих станциях пользователей. См. Глава 4, “Установка программ просмотра Web”, на стр. 45.

Глава 4. Установка программ просмотра Web

Обзор

IBM предоставляет программы просмотра для стандартных документов, загружаемых с сервера OnDemand. Требования к установке зависят от выбранных программ просмотра, которые будут применяться в вашей организации.

- Для просмотра документов строковых данных IBM рекомендует применять апплет Строковые данные. Апплет Строковые данные хранится на сервере HTTP. После разрешения применения апплета Строковые данные он автоматически загружается в память рабочей станции, если пользователь открывает документ строковых данных. Применяемая программа просмотра задается с помощью параметра LINEVIEWING в файле ARSWWW.INI.
- Документы AFP можно просмотреть с помощью IBM OnDemand AFP Web Viewer, апплета AFP2HTML или программы просмотра Adobe Acrobat.
 - Для просмотра документов AFP с помощью программы IBM OnDemand AFP Web Viewer ее необходимо установить на рабочие станции пользователей.
 - Для просмотра документов AFP с помощью апплета AFP2HTML администратор должен установить программу AFP2WEB Transform на сервер HTTP и настроить ее, а также настроить файл ARSWWW.INI. Апплет AFP2HTML хранится на сервере HTTP. После разрешения применения апплета AFP2HTML он автоматически загружается в память рабочей станции, если пользователь открывает документ AFP.
 - Для просмотра документов AFP с помощью программы просмотра Adobe Acrobat администратор должен установить программу AFP2PDF Transform на сервер HTTP и настроить ее, а также настроить файл ARSWWW.INI. После разрешения применения преобразования браузер по умолчанию пытается запустить программу просмотра Adobe Acrobat, если пользователь выбирает просмотр документа AFP. Программа просмотра Adobe Acrobat устанавливается на рабочей станции пользователя.

Применяемая программа просмотра задается с помощью параметра AFPVIEWING в файле ARSWWW.INI.

- Для просмотра документов BMP, GIF, JPEG, PCX и TIFF IBM рекомендует установить на рабочих станциях программу IBM OnDemand Image Web Viewer. В противном случае для работы с этими документами потребуются другие программы просмотра. (Например, в большинство браузеров встроены программы просмотра, позволяющие просматривать документы GIF и JPEG.) Если вы планируете применять другую программу просмотра, убедитесь, что тип данных MIME для этих документов изменен соответствующим образом. Применяемые программы просмотра должны быть указаны с помощью параметров, приведенных в разделе MIMETYPES файла ARSWWW.INI.

Примечания:

1. Для просмотра других типов данных может потребоваться установка дополнительных программ просмотра. Например, для просмотра документов PDF, загружаемых с сервера OnDemand, IBM рекомендует установить программу просмотра Adobe Acrobat для браузеров, применяемых в вашей компании.
2. Для просмотра документов PDF в каталоге встраиваемых модулей браузера должен быть расположен файл nppdf32.dll. Для Internet Explorer - это каталог \Program Files\Internet Explorer\PLUGINS. Для Netscape - каталог \Program

Files\Netscape\Communicator\Program\Plugins. В противном случае потребуется переустановка программного обеспечения Adobe.

Требования

Для работы программ просмотра, предоставленных IBM, необходим Netscape Navigator 4.7 и выше или Microsoft Internet Explorer 5.5 и выше.

В процессе работы ODWEK записывает cookie на компьютер клиента. Убедитесь, что браузеры пользователей настроены таким образом, что cookie принимаются.

IBM предоставляет две версии программы просмотра строковых данных Java, которые расположены в каталоге апплетов:

Файл **ODLineDataViewer.jar** представляет собой старую версию программы просмотра строковых данных Java, для работы которой браузер должен поддерживать Java. Как правило поддержка Java обеспечивается с помощью Виртуальной машины Java (JVM). **Примечание:** При работе с браузером Microsoft Internet Explorer JVM устанавливается с помощью опции Пользовательская установка.

Файл **ODLineDataViewer2.jar** представляет собой новую версию программы просмотра строковых данных Java, для работы которой необходим встраиваемый модуль Java версии 1.4.1 или более поздней. Для работы новой версии программы просмотра строковых данных Java поддержка Java в браузере не требуется. Пользователь должен установить встраиваемый модуль Java на рабочую станцию, чтобы работать с новой программой просмотра строковых данных Java. Важная информация о настройке приведена в разделе “Программа просмотра строковых данных Java” на стр. 52.

По умолчанию вместе с продуктом устанавливается старая версия программы просмотра строковых данных Java.

Браузер должен работать под управлением операционной системы Windows 2000, Windows XP или Windows Server 2003. Кроме того, требуется следующее программное и аппаратное обеспечение:

- Физическое подключение к сети, например, сетевой адаптер Token Ring или Ethernet
- Поддержка протокола TCP/IP Windows
- Минимум 256 Мб оперативной памяти
- Процессор Pentium или совместимый с тактовой частотой 800 МГц и выше
- Монитор SVGA и адаптер, поддерживающий разрешение 800 x 600
- Как минимум 20 Мб свободной дисковой памяти для просмотра документов
- Примерно 3 Мб дисковой памяти на каждой рабочей станции для программы IBM OnDemand AFP Web и 2 Мб для программы IBM OnDemand Image Web Viewer.

Установка

Примечание: Если вы планируете распространять вместе с программой AFP Web Viewer пользовательские файлы, то эти файлы следует включить в установочный файл AFP Web Viewer перед установкой AFP Web Viewer на рабочие станции пользователей. За дополнительной информацией обратитесь к разделу “Распространение пользовательских файлов” на стр. 47.

Программы просмотра, предоставленные IBM, устанавливаются с помощью самораспаковывающихся файлов. Для этого установочные файлы следует загрузить на рабочую станцию Windows пользователя и запустить. Если в процессе установки браузер был открыт, то для применения программы просмотра его следует перезапустить. В каталоге /QIBM/ProdData/OnDemand/www/plugins расположены следующие файлы программ просмотра:

- afpplgus.exe - IBM OnDemand AFP Web Viewer - Только английский язык
- afpplgin.exe - IBM OnDemand AFP Web Viewer - Все языки, включая поддержку DBCS
- afpplgin.zip - IBM OnDemand AFP Web Viewer - Все языки, включая поддержку DBCS, в формате ZIP
- imgplgin.exe - IBM OnDemand Image Web Viewer - Все языки

В процессе установки в каталоги, выбранные пользователем, копируется программа просмотра и связанные с ней файлы. Для установки AFP Web Viewer необходимо примерно 3 Мб дисковой памяти на рабочей станции. Для установки Image Web Viewer потребуется примерно 2 Мб. Напомните пользователям, что если браузер был активен во время установки, его необходимо перезапустить.

Примечание: Программы просмотра устанавливаются в качестве встраиваемых модулей или компонентов ActiveX. Если на рабочей станции установлен Internet Explorer, то устанавливаются компоненты ActiveX. Если установлен Netscape, устанавливаются встраиваемые модули. Если на рабочей станции установлен как Internet Explorer, так и Netscape, то устанавливаются компоненты ActiveX для Internet Explorer и встраиваемые модули для Netscape.

Распространение пользовательских файлов

Программное обеспечение IBM OnDemand AFP Web Viewer, поставляемое IBM, позволяет распространять пользовательские файлы. Например, предположим, что сотрудник вашей организации создал для документов, хранимых на сервере OnDemand, файлы шрифтов AFP. С помощью программного обеспечения AFP Web Viewer вы можете распространить эти файлы шрифтов. Таким образом, документ AFP всегда будет отображаться с помощью правильных шрифтов.

Для распространения пользовательских файлов вместе с программой AFP Web Viewer их необходимо добавить в установочный файл и сохранить в общем каталоге. При запуске установочного файла программа установки автоматически устанавливает на рабочую станцию пользователя программу AFP Web Viewer и пользовательские файлы.

Вместе с программой AFP Web Viewer можно распространять пользовательские файлы следующих типов:

- Файлы шрифтов AFP. Эти файлы копируются в подкаталог FONT целевого каталога программы AFP Web Viewer на рабочей станции.
- Файлы шрифтов Adobe Type 1. Эти файлы копируются в каталог, указанный пользователем, и устанавливаются в ATM программой установки.
- Файлы шрифтов TrueType. Эти файлы копируются в каталог FONTS операционной системы Windows и регистрируются в Windows программой установки.
- Различные пользовательские файлы. Эти файлы копируются в целевой каталог программы AFP Web Viewer на рабочей станции.

Примечание: Программа установки копирует пользовательские файлы на рабочую станцию после файлов программы AFP Web Viewer, предоставленных IBM. Если пользовательскому файлу присвоить имя, такое же как у одного из файлов, предоставленных IBM, то пользовательский файл заменит его. С помощью этой функции вы можете распространять обновленный файл FLDPORT2.INI или файлы шрифтов IBM AFP, измененные в соответствии с потребностями вашей организации.

Дополнительная информация о настройке и распространении AFP Web Viewer проведена в следующих разделах:

- Установка файлов AFP Web Viewer, предоставленных IBM
- Добавление подкаталогов для пользовательских файлов
- Сохранение пользовательских файлов в подкаталогах
- Настройка файлов шрифтов
- Создание установочного файла AFP Web Viewer
- Установка AFP Web Viewer на рабочей станции пользователя.

Установка файлов AFP Web Viewer

Как правило файлы программы просмотра распространяются с сервера двумя способами, в зависимости от того, должны ли вместе с AFP Web Viewer распространяться пользовательские файлы:

- Стандартная установка. Применяется для распространения файлов AFP Web Viewer, предоставленных IBM, а также для подготовки распространения пользовательских файлов вместе с AFP Web Viewer. В процессе установки программного обеспечения ODWEK на сервер HTTP администратор сохраняет установочные файлы в каталоге этого сервера. Среди них для каждой программы просмотра должен быть установочный файл (EXE) и архив ZIP для программы AFP Web Viewer. Как правило администратор перемещает установочные файлы в общий каталог сервера и создает Web-страницу с ссылками на эти файлы. Для установки программы просмотра пользователь открывает Web-страницу в браузере и переходит по ссылке, связанной с соответствующим установочным файлом.
- Пользовательская установка AFP Web Viewer. Позволяет вместе с AFP Web Viewer распространять пользовательские файлы.
 1. Подготовка сервера к стандартной установке.
 2. Перед установкой программы просмотра следует загрузить копию архива ZIP программы AFP Web Viewer.
 3. Извлеките файлы из архива ZIP в пустой рабочий каталог.
 4. Создайте подкаталоги в рабочем каталоге и сохраните в них пользовательские файлы. Более подробная информация приведена в разделах “Создание подкаталогов” на стр. 49 и “Сохранение пользовательских файлов” на стр. 49.
 5. В случае распространения пользовательских файлов шрифтов Adobe Type 1 следует создать файл конфигурации. Более подробная информация приведена в разделе “Настройка файлов шрифтов” на стр. 50.
 6. После настройки всех каталогов и файлов создайте самораспаковывающийся архив EXE, необходимый для распространения. Более подробная информация приведена в разделе “Создание установочного файла AFP Web Viewer” на стр. 50.
 7. Замените файл EXE, предоставленный IBM для стандартной установки, на созданный самораспаковывающийся файл EXE.

8. После того, как администратор выполнит шаги 1 на стр. 48 - 7 на стр. 48, пользователи могут установить AFP Web Viewer вместе с пользовательскими файлами. Для этого необходимо открыть Web-страницу в браузере и перейти по ссылке на обновленный установочный файл.

Создание подкаталогов

Пользовательские файлы, предназначенные для распространения, следует сохранять в подкаталоге CUSTOM основного установочного каталога клиента. Например, основной установочный каталог клиента может называться \ONDEMAND\AFP32.

Для того чтобы настроить основной каталог клиента для хранения пользовательских файлов выполните следующие действия:

1. Создайте каталог CUSTOM в основном установочном каталоге клиента.

Например:

```
\ondemand\afp32\custom
```

Примечание: в каталоге CUSTOM могут храниться другие¹ пользовательские файлы, предназначенные для распространения. Программа установки копирует файлы из этого каталога в целевой каталог программы AFP Web Viewer на рабочей станции.

2. Добавьте следующие подкаталоги в каталог CUSTOM. Название подкаталога зависит от типа пользовательских файлов, предназначенных для распространения пользователям.

- Подкаталог FONT каталога CUSTOM позволяет сохранять файлы шрифтов AFP (FNT и MAP). Например:

```
\ondemand\afp32\custom\font
```

Программа установки копирует файлы из этого каталога в каталог FONT программы AFP Web Viewer на рабочей станции.

- Подкаталог TYPEONE каталога CUSTOM позволяет сохранять файлы шрифтов Adobe Type 1 (PFB и PFM), а также файл конфигурации. Например:

```
\ondemand\afp32\custom\typeone
```

Программа установки копирует эти файлы в каталог, указанный пользователем, и устанавливает шрифты в ATM.

- Подкаталог TRUETYPE каталога CUSTOM позволяет сохранять файлы шрифтов Windows TrueType (TTF). Например:

```
\ondemand\afp32\custom\truetype
```

Программа установки копирует эти файлы в каталог Windows FONT и устанавливает шрифты в операционной системе Windows.

Сохранение пользовательских файлов

После извлечения установочных файлов, предоставленных IBM, в рабочий каталог и создания каталогов CUSTOM пользовательские файлы следует сохранить в отдельных каталогах. Например, скопировать файлы шрифтов Adobe Type 1 (PFB и PFM) в каталог \ONDEMAND\AFP32\CUSTOM\TYPEONE.

1. Кроме файлов шрифтов AFP, Adobe Type 1 и Windows TrueType

Настройка файлов шрифтов

Если вы планируете распространять пользовательские файлы шрифтов Adobe Type, выполните следующие действия:

1. Сохраните пользовательские файлы шрифтов Adobe Type 1 (PFB и PFM) в подкаталог TYPEONE каталога CUSTOM. За дополнительной информацией обратитесь к разделу “Создание подкаталогов” на стр. 49.
2. Создайте файл конфигурации шрифта Type 1. Ниже приведены инструкции по созданию файла конфигурации шрифта Type 1.

Файл конфигурации шрифта Type 1 должен называться ATM_INI.CFG и находиться в подкаталоге TYPEONE каталога CUSTOM. Дополнительная информация о каталогах рассылки приведена в разделе “Создание подкаталогов” на стр. 49.

Каждая запись (строка) файла конфигурации шрифта Type 1 определяет один пользовательский шрифт Adobe Type 1, предназначенный для рассылки пользователям. Для записи применяется следующий формат:

имя-шрифта=имя-файла.PFM, имя-файла.PFB

Где имя-шрифта - это имя шрифта Type 1, появляющееся в списке шрифтов Панели управления ATM, имя-файла.PFM - имя файла PFM шрифта, а имя-файла.PFB - имя файла PFB шрифта. Ниже приведен пример файла конфигурации шрифта Type 1 с двумя записями:

```
Courier,BOLD=coub.pfm,coub.pfb  
SonoranSansSerif_36,BOLDITALIC=c0a175z0.pfm,c0a175z0.pfb
```

Первая запись в этом файле определяет шрифт Courier,BOLD, файл шрифта PFM coub.pfm и файл шрифта PFB coub.pfb. Вторая запись определяет шрифт SonoranSansSerif,BOLDITALIC, файл шрифта PFM c0a175z0.pfm и файл шрифта PFB c0a175z0.pfb.

При запуске установочного файла AFP Web Viewer, содержащего пользовательские файлы шрифтов Adobe Type 1, программа установки обрабатывает их следующим образом:

1. Копирует пользовательские файлы шрифтов Adobe Type 1 (PFB и PFM) из каталога TYPEONE в целевой каталог. Целевой каталог выбирает пользователь.
2. Проверяет, что были скопированы два файла для каждого шрифта, указанного в файле конфигурации Type 1 (ATM_INI.CFG). Имена файлов, скопированных на рабочую станцию, должны совпадать с именами, указанными в файле конфигурации шрифтов.

Примечание: Если имена файлов шрифтов из файла конфигурации не совпадают с именами файлов, скопированных на рабочую станцию, то программа установки выдает предупреждающее сообщение и не устанавливает шрифт.

3. Добавляет информацию о путях к файлам PFB и PFM с помощью целевого каталога, указанного пользователем.
4. Устанавливает шрифты в ATM.

Создание установочного файла AFP Web Viewer

После создания каталогов и сохранения файлов в каталоге CUSTOM следует создать установочный файл, содержащий пользовательские файлы, а также файлы AFP Web Viewer, предоставленные IBM. Как правило, установочный файл называется Setup.exe.

Некоторые компании выпускают программное обеспечение для упаковки файлов и приложений в один самораспаковывающийся файл AFP Web Viewer, предназначенный для распространения. Например, компания InstallShield Software Corporation предлагает продукт под названием PackageForTheWeb.

Примечание: Программное обеспечение других компаний не поддерживается IBM.

С помощью соответствующего программного обеспечения создайте установочный файл AFP Web Viewer, содержащий пользовательские файлы и файлы AFP Web Viewer, предоставленные IBM.

Установка AFP Web Viewer на рабочей станции пользователя

После создания каталога CUSTOM создайте установочный файл AFP Web Viewer и замените его на сервере. При этом пользователи смогут начать установку AFP Web Viewer и пользовательских файлов. Следующее открытие ссылки на установочный файл AFP Web Viewer приведет к установке AFP Web Viewer на рабочую станцию пользователя и копированию на нее пользовательских файлов, добавленных в установочный файл AFP Web Viewer.

Преобразование шрифтов AFP

Шрифты AFP, с помощью которых создан документ, должны быть связаны со шрифтами, отображаемыми с помощью встраиваемого модуля AFP. В программном обеспечении ODWEK предусмотрены файлы определения шрифтов, позволяющие связать Базовые шрифты IBM для обмена информацией (только латинские) и Совместимые шрифты со шрифтами TrueType. Файлы определения и преобразования шрифтов распложены в подкаталоге FONT, в котором хранится исходный код программы AFP Web Viewer.

Если в документах применяются шрифты, не определенные в программе AFP Web Viewer, если были изменены Базовые шрифты IBM или созданы шрифты AFP, то для правильного отображения документов с помощью программы AFP Web Viewer следует определить эти шрифты в файлах определения шрифтов. Инструкции по преобразованию шрифтов AFP, описание файлов определения шрифтов и другая техническая информация, связанная со шрифтами AFP и TrueType, приведена в разделе *AFPWorkbench Technical Reference*.

AFP Web viewer

Следующие параметры логических представлений можно применить на сервере для программы AFP Web Viewer.

- Цвет фона. Поддерживаются следующие цвета. Другие цвета не предусмотрены.
 - Стандартный зеленый (отображается на белом фоне)
 - Зеленый
 - Красный
 - Желтый
 - Черный
 - Белый
 - Серый
- Цвет изображения. Поддерживаются следующие цвета. Другие цвета не предусмотрены.

- | Желтый
- | Синий
- | Красный
- | Лиловый
- | Зеленый
- | Бирюзовый
- | По умолчанию (отображается черным цветом)
- | • Масштаб.

| **Примечание:** Цвет выбранной области в программе AFP Web Viewer не применяется.
| Для выбранной области всегда применяется белый цвет для текста и
| черный цвет для фона.

| Image Web viewer

| Следующая информация необходима для просмотра многостраничных изображений
| с помощью программы Image Web Viewer.

| **Примечание:** В ходе выполнения следующей процедуры вносятся изменения в реестр
| компьютера. Изменять реестр следует только в случае крайней
| необходимости. Ошибки реестра могут привести к неправильной работе
| компьютера. Перед тем как продолжить следует создать резервную
| копию реестра, а также ознакомиться с инструкциями по
| восстановлению версии реестра, применявшейся после последнего
| успешного запуска компьютера. Инструкции приведены в справке
| операционной системы Windows.

| При работе с многостраничными изображениями, если перетаскивается маркер
| вертикальной полосы прокрутки, то рядом с ним отображается небольшое окно. В
| этом окне показан номер страницы, соответствующий положению маркера и числу
| страниц изображения. Например, значение 5 / 10 указывает, что изображение
| состоит из десяти страниц и, если отпустить маркер, то пятая страница станет
| текущей.

| Для отмены этой функции укажите в реестре следующий ключ:

| HKEY_LOCAL_MACHINE\Software\IBM\OnDemand Image Web Viewer\Preferences

| Если для строкового параметра PageNumberScroll задано значение 0 (ноль), то при
| перемещении маркера полосы прокрутки окно номера страницы отображаться не
| будет.

| Если в этом ключе реестра для строкового параметра PageNumberToolbar задано
| значение 1, то для многостраничных изображений будет отображаться информация о
| номере страницы в панели инструментов. Например, значение 3 / 5 указывает, что
| изображение состоит из пяти страниц и третья страница является текущей.

| Программа просмотра строковых данных Java

| IBM предоставляет расширенную версию программы просмотра строковых данных
| Java. В число улучшений входят расширенные функции печати, такие как печать
| страницы по всей ширине. Графический пользовательский интерфейс реализован на
| основе библиотеки Swing.

IBM предоставляет две версии программы просмотра строковых данных Java, которые расположены в каталоге апплетов:

Файл **ODLineDataViewer.jar** представляет собой старую версию программы просмотра строковых данных Java, для работы которой необходим встраиваемый модуль Java версии 1.1.8 или более поздних версий.

Файл **ODLineDataViewer2.jar** представляет собой новую версию программы просмотра строковых данных Java, для работы которой необходим встраиваемый модуль Java версии 1.4.1 или более поздних версий.

Заказчики могут работать как с новой версией программы просмотра строковых данных Java, так и со старой версией. Для выбора необходимой версии укажите соответствующие параметры в разделе [DEFAULT BROWSER] файла ARSWWW.INI. Кроме того, для работы новой версии программы просмотра строковых данных Java для браузера следует установить встраиваемый модуль Java версии 1.4.1. Дополнительные параметры, указанные в файле ARSWWW.INI, определяют номер версии и расположение установочного файла встраиваемого модуля Java для тех пользователей, на рабочих станциях которых не установлена необходимая версия встраиваемого модуля Java.

В разделе Табл. 2 описаны новые параметры файла ARSWWW.INI, поддерживающие программу просмотра строковых данных Java.

Таблица 2. Параметры файла ARSWWW.INI, предусмотренные для программы просмотра строковых данных Java

Параметр	Значение	Комментарии
ODApplet.version	1	Позволяет вызвать старую версию программы просмотра строковых данных Java. Если этот параметр указан, то остальные параметры игнорируются. Примечание: Это значение по умолчанию. Если этот параметр пропущен, то ODWEK применяет старую версию программы просмотра строковых данных Java.
	2	Позволяет вызвать новую версию программы просмотра строковых данных Java (расширенная версия). Если этот параметр указан, то необходимо указать следующие три параметра.

Таблица 2. Параметры файла ARSWWW.INI, предусмотренные для программы просмотра строковых данных Java (продолжение)

Параметр	Значение	Комментарии
ODApplet.jre.path.IE	http://java.sun.com/getjava/installer.html	Для браузера Internet Explorer. Позволяет автоматически загрузить и установить последнюю версию встраиваемого модуля Java с Web-сайта java.sun.com. Для предварительного просмотра результатов автоматической загрузки и установки встраиваемого модуля Java откройте страницу http://java.sun.com/getjava/install-windows.html . Примечание: После установки встраиваемого модуля может потребоваться перезагрузка компьютера.
	<location>	Задаёт расположение установочного файла встраиваемого модуля Java в локальной сети компании. Расположение должно включать в себя допустимый протокол браузера, такой как http, file или ftp. Например: file://shareName/java/plugins/plugin.exe Примечание: Установочный файл встраиваемого модуля Java должен быть загружен и сохранен в указанном расположении системным администратором. Если расположение установочного файла задано, то браузер автоматически установит соответствующий встраиваемый модуль Java на рабочей станции. После установки встраиваемого модуля может потребоваться перезагрузка браузера.

Таблица 2. Параметры файла ARSWWW.INI, предусмотренные для программы просмотра строковых данных Java (продолжение)

Параметр	Значение	Комментарии
ODApplet.jre.path.NN	http://java.sun.com/j2se/1.4.1/download.html	Для браузера Netscape. Позволяет открыть страницу загрузки JRE/J2SE для выбора устанавливаемого встраиваемого модуля Java. Для загрузки установочного файла встраиваемого модуля Java для необходимой платформы пользователь должен перейти по соответствующей ссылке. После загрузки установочного файла встраиваемого модуля Java его следует установить на рабочую станцию. После установки встраиваемого модуля может потребоваться перезагрузка браузера.
	<location>	<p>Задаёт расположение файлов встраиваемого модуля в локальной сети компании. Расположение должно включать в себя допустимый протокол браузера, такой как http, file или ftp. Например:</p> <p>http://webServer/tmp/ondemand/java/plugins</p> <p>Примечание: Файлы встраиваемого модуля должны быть загружены и сохранены в указанном расположении системным администратором. Путь к конкретному файлу указать нельзя, так как неизвестно под управлением какой операционной системы работает браузер Netscape. Кроме того, указанный формат позволяет администратору при необходимости загрузить встраиваемый модуль для других платформ.</p> <p>Пользователь должен установить встраиваемый модуль Java на рабочую станцию. После установки встраиваемого модуля может потребоваться перезагрузка браузера.</p>

Таблица 2. Параметры файла ARSWWW.INI, предусмотренные для программы просмотра строковых данных Java (продолжение)

Параметр	Значение	Комментарии
ODApplet.jre.version	<version>	<p>Задаёт версию применяемого встраиваемого модуля Java. Следует указать версию 1.4 или более позднюю. Укажите основной номер версии (например, 1.4) для поддержки выпусков встраиваемого модуля этого уровня (например, 1.4.0, 1.4.0_03, 1.4.1_01). Укажите конкретный номер версии (например, 1.4.1_01) для поддержки встраиваемого модуля Java только данной версии. Допустимые номера версий можно получить на Web-сайте java.sun.com.</p> <p>Например:</p> <p>1.4</p> <p>или:</p> <p>1.4.1_01</p>

В следующем примере описывается, каким образом можно настроить файл ARSWWW.INI для поддержки старой версии программы просмотра строковых данных Java.

```
[DEFAULT BROWSER]
ODApplet.version=1
```

Примечания:

1. Если не указать параметр ODApplet.version в файле ARSWWW.INI, то ODWEK будет использовать старую версию программы просмотра строковых данных Java.
2. Параметры ODApplet являются глобальными и их можно указывать только в разделе DEFAULT BROWSER. (Если указать эти параметры в других разделах, то они будут проигнорированы.)

Ниже приведен пример файла ARSWWW.INI, настроенного для поддержки новой программы просмотра строковых данных Java (расширенная версия), а также встраиваемого модуля Java версии 1.4 или более поздней. Браузер Internet Explorer позволяет автоматически загрузить и установить последнюю версию встраиваемого модуля Java с Web-сайта java.sun.com. В случае Netscape администратор сохраняет установочные файлы встраиваемого модуля Java для различных платформ на локальном Web-сервере. При этом не требуется загружать файлы со страницы загрузки JRE/J2SE Web-сайта java.sun.com. **Примечание:** Загрузка и установка встраиваемого модуля Java требуется только для тех рабочих станций, на которых не установлена версия 1.4 или более поздняя версия этого модуля.

```
[DEFAULT BROWSER]
ODApplet.version=2
ODApplet.jre.path.IE=http://java.sun.com/getjava/installer.html
ODApplet.jre.path.NN=http://localWebServer/java/plugins
ODApplet.jre.version=1.4
```

Следующий шаг

После установки программного обеспечения ODWEK, настройки файла ARSWWW.INI, настройки примеров приложений и установки программ просмотра Web программное обеспечение ODWEK готово к работе.

Приложение А. Справочник по API CGI

В этом разделе приведена информация о функциях программирования, доступных в ODWEK. Этот раздел предназначен для программистов, ответственных за интегрирование ODWEK в Web-браузеры.

Примечание: Значения параметров представляют собой стандартный текст. Существует вероятность того, что браузер не сможет распознать некоторые символы текста. Для предотвращения возможных ошибок все специальные символы следует указывать в виде соответствующих шестнадцатеричных кодов. В состав специальных символов входят символы управления и определенные буквенно-цифровые символы. Например, строка:

The post date is 12/31/95

будет преобразована следующим образом:

The%20post%20date%20is%2012%2f31%2f95

Значениями параметров могут быть имена папок, имена полей папок, а также критерии поиска.

Добавление комментариев

Добавление комментария в указанный документ

Цель

Функция **Добавить комментарий** позволяет добавить комментарии в указанный документ. Для того чтобы добавить комментарий пользователь должен обладать правами доступа на добавление комментариев для группы приложений OnDemand. (Права доступа к группе приложений позволяют добавлять комментарии.)

Параметры

Таблица 3. функция **Добавить комментарий**

Имя=значение	Назначение
_function=addnote	Добавление комментария.
_server=значение	Имя сервера OnDemand.
_user=value	ИД пользователя OnDemand. Для того чтобы добавить комментарий пользователь должен обладать правами доступа на добавление комментариев для каждой группы приложений, в которых содержатся необходимые документы. (Права доступа к группе приложений позволяют добавлять комментарии.)
_password=значение	Пароль пользователя
_folder=значение	Имя папки.
_perm=значение	Определяет права доступа к комментарию: Общий (0), Личный (1) или Личный для группы (2). Общие комментарии доступны для всех пользователей, обладающих правами доступа на просмотр комментариев для данной группы приложений. Личные комментарии могут просматривать только пользователи, создавшие их, администраторы группы приложений и системные администраторы. Личные комментарии группы могут просматривать только пользователи, входящие в состав этой группы, администраторы группы приложений и системные администраторы. В параметре _group содержится имя группы. Значение по умолчанию - 0 (Общая).
_group=имя-группы	Задает имя группы, если в параметре _perm указано значение 2 (Личный для группы).
_copy=значение	Позволяет указать, следует ли при экспорте документа на другой сервер оставлять комментарий, прикрепленным к нему. Значение по умолчанию off. Оно означает, что комментарий к документу не добавляется. Значение on означает, что комментарий добавляется к документу в случае его экспорта на другой сервер.
_text=значение	Текст комментария.

Таблица 3. функция *Добавить комментарий* (продолжение)

Имя=Значение	Назначение
_html=значение	<p>Определяет файл HTML, применяемый ODWEK в качестве шаблона для создания Web-страниц вывода. В качестве значения можно указать имя файла или символ *. Если указана звездочка, то применяется файл ADDNOTE.HTML, расположенный в каталоге, заданном с помощью параметра TEMPLATEDIR в файле ARSWWW.INI. Если указано имя файла без пути, то файл должен быть расположен в каталоге, заданном в параметре TEMPLATEDIR. Если вместе с именем файла указан путь к нему, он задается относительно каталога, указанного в параметре TEMPLATEDIR.</p> <p>Общий формат файла HTML определяется пользователем. Однако файл шаблона должен содержать следующую строку комментария:</p> <pre><!-- -AOI# Marker-- -></pre> <p>Расположение строки комментария определяет расположение вывода ODWEK. Строки, указанные до строки комментария, отображаются до вывода ODWEK. Строки, указанные после строки комментария, отображаются после вывода ODWEK.</p> <p>Файл TEMPLATE.HTM представляет собой пример файла шаблона, поставляемого вместе с программным обеспечением ODWEK. На основе примера файла шаблона вы можете создать собственный файл шаблона, позволяющий добавить функцию работы с комментариями.</p>
_nohtml=значение	<p>Определяет тип вывода ODWEK. Значение по умолчанию - 0 (ноль). Оно означает, что создается вывод HTML. Если указано значение 1, то ODWEK создает вывод ASCII с ограничителями. Дополнительная информация о выводе ASCII с ограничителями в разделе Приложение Н, "Не создавать вывод HTML", на стр. 137.</p>
_docid=ИД документа	<p>Идентификатор документа, к которому добавляется комментарий. Идентификатор документа возвращается функцией Список найденных документов.</p>
_port=значение	<p>Номер порта сервера OnDemand. Значение по умолчанию - 0 (ноль). Оно означает, что применяется порт, номер которого указан в Таблице служб (WRKSRVTBLE). Если номер порта не указан в Таблице служб, то по умолчанию применяется порт 1445. Указанное значение переопределяет значение параметра PORT из файла ARSWWW.INI.</p>
_codepage=значение	<p>Задает кодовую страницу базы данных OnDemand. По умолчанию применяется кодовая страница сервера HTTP. Кодовую страницу следует указывать, если кодовая страница сервера отличается от кодовой страницы базы данных. Указанное значение переопределяет значение параметра CODEPAGE из файла ARSWWW.INI.</p>
_logoff=1	<p>Автоматически прерывает связь рабочей станции пользователя с сервером после добавления комментария. Данный параметр позволяет избежать необходимости вызова функции Выход из системы для отключения пользователя. Для этого параметра допустимо только значение 1.</p>

Использование

Следующие параметры являются обязательными:

- _function
- _server
- _user
- _password
- _text
- _docid

Следующие параметры являются необязательными:

`_perm`
`_group` (необходим, если в параметре `if _perm` указано значение Личный для группы)
`_html`
`_nohtml`
`_port`
`_codepage`
`_logoff`

Пример вызова функции

```
http://www.company.com/cgi-bin/arswww.cgi?_function=addnote  
&_server=od400&_user=web&_password=web  
&_folder=credit%20card%20statements  
&_text=Test%20note%20from%20the%20OnDemand%20Internet%20Client  
&_docid=6850-6851-SUA17-1FAAA-225712-1634-132014-132172-89-76-11-25-0  
&_perm=1&_logoff=1
```

Изменить пароль

Изменить пароль входа на сервер OnDemand

Цель

Функция Изменить пароль позволяет пользователям изменять свои пароли OnDemand.

Параметры

Таблица 4. функция Изменить пароль

Имя=Значение	Назначение
_function=chgpassword	Позволяет изменить пароль OnDemand для указанного ИД пользователя.
_server=значение	Имя сервера OnDemand.
_user= value	ИД пользователя OnDemand.
_password= значение	Пароль для ИД пользователя.
_new_password=значение	Новый пароль для ИД пользователя.
_html=значение	<p>Определяет файл HTML, применяемый ODWEK в качестве шаблона для создания Web-страниц вывода. В качестве значения можно указать имя файла или символ *. Если указана звездочка, то применяется файл CHGPASSWORD.HTML, расположенный в каталоге, заданном с помощью параметра TEMPLATEDIR в файле ARSWWW.INI. Если указано имя файла без пути, то файл должен быть расположен в каталоге, заданном в параметре TEMPLATEDIR. Если вместе с именем файла указан путь к нему, он задается относительно каталога, указанного в параметре TEMPLATEDIR.</p> <p>Общий формат файла HTML определяется пользователем. Однако файл шаблона должен содержать следующую строку комментария:</p> <pre><!-- - -AOI# Marker-- - -></pre> <p>Расположение строки комментария определяет расположение вывода ODWEK. Строки, указанные до строки комментария, отображаются до вывода ODWEK. Строки, указанные после строки комментария, отображаются после вывода ODWEK.</p> <p>Файл TEMPLATE.HTM представляет собой пример файла шаблона, поставляемого вместе с программным обеспечением ODWEK. На основе примера файла шаблона вы можете создать собственный файл шаблона для функции изменения пароля.</p>
_nohtml= значение	Определяет тип вывода ODWEK. Значение по умолчанию - 0 (ноль). Оно означает, что создается вывод HTML. Если указано значение 1, то ODWEK создает вывод ASCII с ограничителями. Дополнительная информация о выводе ASCII с ограничителями в разделе Приложение Н, "Не создавать вывод HTML", на стр. 137.
_port= значение	Номер порта сервера OnDemand. Значение по умолчанию - 0 (ноль). Оно означает, что применяется порт, номер которого указан в Таблице служб (WRKSRVTBLE). Если номер порта не указан в Таблице служб, то по умолчанию применяется порт 1445. Указанное значение переопределяет значение параметра PORT из файла ARSWWW.INI.
_codepage= значение	Задаёт кодировку страницы базы данных OnDemand. По умолчанию применяется кодировка страницы сервера HTTP. Кодовую страницу следует указывать, если кодировка страницы сервера отличается от кодировки страницы базы данных. Указанное значение переопределяет значение параметра CODEPAGE из файла ARSWWW.INI.

Таблица 4. функция Изменить пароль (продолжение)

Имя=Значение	Назначение
<code>_cgibin=программа</code>	<p>Применяется программой CGI для создания следующей страницы вывода. Если этот параметр указан, то вместо программы по умолчанию (ARSWWW.CGI) в странице будет содержаться вызов заданной программы. Как правило, этот параметр применяется программистами, разрабатывающими клиентские программы CGI или сервлеты для программ CGI или сервлетов, предоставленных IBM.</p> <p>В качестве <i>программы</i> можно указать каталог, относительно директивы ServerRoot, или <i>псевдоним</i>, определенный в файле конфигурации сервера HTTP. По умолчанию ODWEK загружает программу CGI из каталога CGI-BIN.</p>
<code>_logoff=1</code>	<p>Автоматически отключает пользователя от сервера после изменения пароля. Данный параметр позволяет избежать необходимости вызова функции Выход из системы для отключения пользователя. Для этого параметра допустимо только значение 1.</p>

Использование

Следующие параметры являются обязательными:

```
_function
_server
_user
_password
_new_password
```

Следующие параметры являются необязательными:

```
_html
_nohtml
_port
_codepage
_logoff
_cgibin
```

Пример вызова функции

```
http://www.company.com/cgi-bin/arswww.cgi?_function=chgpassword
&_server=od400&_user=web&_password=web
&_newpassword=newpw&_html=template.htm&_logoff=1
```

Список найденных документов

Отображает список документов, соответствующих критериям поиска

Цель

Функция Список найденных документов отображает список документов, соответствующих критериям поиска конкретной папки. Каждый документ представляется в виде ссылки на документ сервера OnDemand. Щелкните на документе для того, чтобы загрузить его с сервера и просмотреть в окне браузера с помощью соответствующей программы просмотра.

Параметры

Таблица 5. Функция Список найденных документов

Имя=Значение	Назначение
_function=dochitlist	Отображает список документов, соответствующих критериям поиска.
_server= значение	Имя сервера OnDemand.
_user=значение	ИД пользователя OnDemand.
_password= значение	Пароль для ИД пользователя.
_folder= значение	Имя папки.
folder field name=значение	Имя поля поиска папки и поисковое значение. Вы можете указать несколько наборов имен полей и поисковых значений, не превышающих число полей, определенных для папки.
folder field name2=значение	Задаёт верхнее значение поиска для полей поиска, в которых применяются операции BETWEEN и NOT BETWEEN.
folder field nameOP=значение	Задаёт оператор для переопределения оператора по умолчанию для поля поиска папки. Предусмотрены следующие значения: 1 Равно 2 Не равно 4 Меньше 8 Меньше или равно 16 Больше 32 Больше или равно 64 Входит в состав 128 Не входит в состав 256 Похоже на 512 Не похоже на 1024 В диапазоне 2048 Не в диапазоне
_display_fields=значение[значение,...]	Список имен полей папки, разделенный запятыми, для отображения. Вы можете указать несколько имен полей. Если этот параметр не указан, то на конечной странице будут перечислены все отображаемые поля папки.
_sort_field=значение[значение,...]	Определяет поле поиска папки, применяемое OnDemand для сортировки элементов списка документов. Вы можете указать несколько полей, разделенных запятыми. Например: _sort_field=Account,Account+Balance,Date . Поля сортировки по умолчанию определены на странице Информация о поле указанной папки.
_sort_order=значение[значение,...]	Позволяет определить порядок сортировки для каждого поля поиска папки, указанного в параметре sort_field . Для сортировки по возрастанию укажите A . Для сортировки по убыванию укажите любой другой символ. Например: _sort_order=A,D,A . Порядок сортировки по умолчанию определяется порядком сортировки, заданным на странице Информация о поле указанной папки.

Таблица 5. Функция Список найденных документов (продолжение)

Имя=Значение	Назначение
_max_hits=значение	<p>Задает максимальное число элементов, отображаемых в списке документов. Данное значение не зависит от числа совпадений. В список документов добавляются элементы, соответствующие критериям запроса, в порядке их загрузки в базу данных.</p> <p>С помощью первого указанного значения определяется число элементов, возвращаемых в список документов:</p> <ol style="list-style-type: none"> 1. Значение поля Максимальное число совпадений (указанное на странице Права доступа к папке). Это значение переопределяет все остальные значения. 2. Значение параметра _max_hits, если он указан. Это значение переопределяет параметр MAXHITS из файла ARSWWW.INI. 3. Значение параметра MAXHITS, если оно указано. 4. Если не указано ни одно из приведенных выше значений, то по умолчанию в список состоит из 200 элементов.
_html= значение	<p>Определяет файл HTML, применяемый ODWEK в качестве шаблона для создания Web-страниц вывода. В качестве значения можно указать имя файла или символ *. Если указана звездочка, то применяется файл DOCHITLIST.HTML, расположенный в каталоге, заданном с помощью параметра TEMPLATEDIR в файле ARSWWW.INI. Если указано имя файла без пути, то файл должен быть расположен в каталоге, заданном в параметре TEMPLATEDIR. Если вместе с именем файла указан путь к нему, он задается относительно каталога, указанного в параметре TEMPLATEDIR.</p> <p>Общий формат файла HTML определяется пользователем. Однако файл шаблона должен содержать следующую строку комментария:</p> <pre><!-- -AOI# Marker-- -></pre> <p>Расположение строки комментария определяет расположение вывода ODWEK. Строки, указанные до строки комментария, отображаются до вывода ODWEK. Строки, указанные после строки комментария, отображаются после вывода ODWEK.</p> <p>Файл TEMPLATE.HTM представляет собой пример файла шаблона, поставляемого вместе с программным обеспечением ODWEK. На основе примера файла шаблона вы можете создать собственный файл шаблона для функции списка документов.</p>
_frame=значение	<p>В вывод этой команды включается атрибут target=значение. Данный параметр упрощает процесс создания фреймов HTML. Это необязательный параметр.</p>
_datefmt=значение	<p>Определяет формат даты для поиска в базе данных и отображения элементов, соответствующих запросу. Формат даты по умолчанию устанавливается на странице Информация о поле папки. Более подробная информация о форматах даты, поддерживаемых OnDemand приведена в книге <i>Руководство администратора общего сервера IBM Content Manager OnDemand for iSeries V5R1</i>, SC27-1161.</p>
_nohtml= значение	<p>Определяет тип вывода ODWEK. Значение по умолчанию - 0 (ноль). Оно означает, что создается вывод HTML. Если указано значение 1, то ODWEK создает вывод ASCII с ограничителями. Дополнительная информация о выводе ASCII с ограничителями в разделе Приложение Н, "Не создавать вывод HTML", на стр. 137.</p>

Таблица 5. Функция Список найденных документов (продолжение)

Имя=Значение	Назначение
_port= значение	Номер порта сервера OnDemand. Значение по умолчанию - 0 (ноль). Оно означает, что применяется порт, номер которого указан в Таблице служб (WRKSRVTBLE). Если номер порта не указан в Таблице служб, то по умолчанию применяется порт 1445. Указанное значение переопределяет значение параметра PORT из файла ARSWWW.INI.
_codepage= значение	Задаёт кодовую страницу базы данных OnDemand. По умолчанию применяется кодовая страница сервера HTTP. Кодовую страницу следует указывать, если кодовая страница сервера отличается от кодовой страницы базы данных. Указанное значение переопределяет значение параметра CODEPAGE из файла ARSWWW.INI.
_sql= строка	<p>Задаёт запрос SQL, применяемый сервером OnDemand для поиска в папке. Если этот параметр указан, то для поиска вместо заданных пар имя/значение поля папки применяется запрос SQL. Сервер OnDemand не проверяет строку запроса.</p> <p>Для работы со строками SQL необходимо указать имена и значения полей базы данных группы приложений. Если вы планируете выполнять поиск в полях данных, необходимо указать внутренние значения даты OnDemand. Например, дата 1 января 1999 будет указана как 10593. Для просмотра внутренних значений для указанной даты воспользуйтесь командой ARSDATE.</p> <p>Строка SQL позволяет выполнить поиск во всех группах приложений, расположенных в папке. Если в строке SQL указано имя поля базы данных, определенное только в одной группе приложений, то запрос выполнен не будет.</p>
_date1= значение	Позволяет указать начальную дату диапазона дат для поиска. Если указаны параметры _date1 и _date2 , то поиск ограничивается одной или несколькими таблицами, в которых содержится одно или оба указанных значения. Формат указанной строки даты должен соответствовать формату отображения поля папки. (Для просмотра формата отображения поля папки воспользуйтесь клиентом администрирования.)
_date2= значение	Позволяет указать конечную дату в диапазоне дат для поиска. Если указаны параметры _date1 и _date2 , то поиск ограничивается одной или несколькими таблицами, в которых содержится одно или оба указанных значения. Формат указанной строки даты должен соответствовать формату отображения поля папки. (Для просмотра формата отображения поля папки воспользуйтесь клиентом администрирования.)
_cgibin= программа	<p>Применяется программой CGI для создания следующей страницы вывода. Если этот параметр указан, то вместо программы по умолчанию (ARSWWW.CGI) в странице будет содержаться вызов заданной программы. Как правило, этот параметр применяется программистами, разрабатывающими клиентские программы CGI или сервлеты для программ CGI или сервлетов, предоставленных IBM.</p> <p>В качестве программы можно указать каталог, относительно директивы ServerRoot, или псевдоним, определенный в файле конфигурации сервера HTTP. По умолчанию ODWEK загружает программу CGI из каталога /QIBM/Proddata/OnDemand/www/bin.</p>
_or= значение	Если указано значение 1, то поля поиска объединяются с помощью логического оператора ИЛИ. При этом документ должен соответствовать по крайней мере одному из указанных значений поиска. Значение по умолчанию - 0 (ноль). Оно означает, что поля поиска объединяются с помощью оператора И (документ должен соответствовать всем указанным значениям поиска).

Таблица 5. Функция Список найденных документов (продолжение)

Имя=Значение	Назначение
_logoff=1	Автоматически отключает пользователя от сервера после создания списка документов. Данный параметр позволяет избежать необходимости вызова функции Выход из системы для отключения пользователя. Для этого параметра допустимо только значение 1.

Использование

Следующие параметры являются обязательными:

`_function`
`_server`
`_user`
`_password`
`_folder`

Следующие параметры являются необязательными:

folder field name
folder field name2
folder field nameOP
`_display_fields`
`_sort_field`
`_sort_order`
`_max_hits`
`_frame`
`_datefmt`
`_sql`
`_date1`
`_date2`
`_or`
`_html`
`_nohtml`
`_port`
`_codepage`
`_logoff`
`_cgibin`

Пример вызова функции

```
http://www.company.com/cgi-bin/arswww.cgi?_function=dochitlist
&_server=od400&_user=web&_password=web
&_folder=credit%20card%20statements
&account%20number=1000100010009999&date=1%2f1%2f96&date2=12%2f31%2f96
&nameOP=256&name=%AA
&_sort_field=Account,Account%20Balance,Date&_sort_order=A,D,A
&_logoff=1
&_html=template.htm
```

Выход из системы

Выход из сервера OnDemand

Цель

Функция Выход из системы позволяет завершить сеанс работы с сервером OnDemand. Имя сервера, а также ИД пользователя сохраняются в cookie браузера клиента функцией Вход в систему. Если указан недопустимый сервер OnDemand, то выдается сообщение об ошибке. Если пользователь не зарегистрирован в указанном сервере OnDemand, выдается сообщение об ошибке.

Параметры

Таблица 6. функция Выход из системы

Имя=Значение	Назначение
<code>_function=logoff</code>	Выход из сервера OnDemand.
<code>_html=значение</code>	<p>Определяет файл HTML, применяемый ODWEK в качестве шаблона для создания Web-страниц вывода. В качестве значения можно указать имя файла или символ *. Если указана звездочка, то применяется файл LOGOFF.HTML, расположенный в каталоге, заданном с помощью параметра <code>TEMPLATEDIR</code> в файле <code>ARSWWW.INI</code>. Если указано имя файла без пути, то файл должен быть расположен в каталоге, заданном в параметре <code>TEMPLATEDIR</code>. Если вместе с именем файла указан путь к нему, он задается относительно каталога, указанного в параметре <code>TEMPLATEDIR</code>.</p> <p>Общий формат файла HTML определяется пользователем. Однако файл шаблона должен содержать следующую строку комментария:</p> <pre><!-- - -AOI# Marker-- - -></pre> <p>Расположение строки комментария определяет расположение вывода ODWEK. Строки, указанные до строки комментария, отображаются до вывода ODWEK. Строки, указанные после строки комментария, отображаются после вывода ODWEK.</p> <p>Файл <code>TEMPLATE.HTM</code> представляет собой пример файла шаблона, поставляемого вместе с программным обеспечением ODWEK. На основе примера файла шаблона вы можете создать собственный файл шаблона для функции выхода из системы.</p>
<code>_nohtml= значение</code>	Определяет тип вывода ODWEK. Значение по умолчанию - 0 (ноль). Оно означает, что создается вывод HTML. Если указано значение 1, то ODWEK создает вывод ASCII с ограничителями. Дополнительная информация о выводе ASCII с ограничителями в разделе Приложение Н, "Не создавать вывод HTML", на стр. 137.
<code>_port= значение</code>	Номер порта сервера OnDemand. Значение по умолчанию - 0 (ноль). Оно означает, что применяется порт, номер которого указан в Таблице служб (<code>WRKSRVTBLE</code>). Если номер порта не указан в Таблице служб, то по умолчанию применяется порт 1445. Указанное значение переопределяет значение параметра <code>PORT</code> из файла <code>ARSWWW.INI</code> .

Использование

Следующие параметры являются обязательными:

`_function`

Следующие параметры являются необязательными:

`_html`

| _nohtml
| _port

Пример вызова функции

`http://www.company.com/cgi-bin/arswww.cgi?_function=logoff
&_html=template.htm`

Вход в систему

Вход на сервер OnDemand

Цель

Функция Вход в систему позволяет получить доступ к серверу OnDemand с помощью следующих параметров: имя сервера, ИД пользователя и пароль. Функция Вход в систему проверяет права доступа указанного пользователя и пароль. Если пользователь не обладает правами доступа к указанному серверу OnDemand, выдается сообщение об ошибке. Если указан недопустимый сервер OnDemand, то выдается сообщение об ошибке. Если указан недопустимый пароль, то выдается сообщение об ошибке. В случае успешного входа в систему отображается Web-страница, в которой приведен список папок, доступных для пользователя.

Параметры

Таблица 7. функция Вход в систему

Имя=Значение	Назначение
<code>_function=logon</code>	Вход на сервер OnDemand.
<code>_server= значение</code>	Имя сервера OnDemand.
<code>_user= значение</code>	ИД пользователя OnDemand.
<code>_password= значение</code>	Пароль для ИД пользователя.
<code>_new_password= значение</code>	Новый пароль для ИД пользователя. Позволяет изменить пароль после успешного входа на сервер OnDemand. Это необязательный параметр.
<code>_html= значение</code>	<p>Определяет файл HTML, применяемый ODWEK в качестве шаблона для создания Web-страниц вывода. В качестве значения можно указать имя файла или символ *. Если указана звездочка, то применяется файл LOGON.HTML, расположенный в каталоге, заданном с помощью параметра <code>TEMPLATEDIR</code> в файле <code>ARSWWW.INI</code>. Если указано имя файла без пути, то файл должен быть расположен в каталоге, заданном в параметре <code>TEMPLATEDIR</code>. Если вместе с именем файла указан путь к нему, он задается относительно каталога, указанного в параметре <code>TEMPLATEDIR</code>.</p> <p>Общий формат файла HTML определяется пользователем. Однако файл шаблона должен содержать следующую строку комментария:</p> <pre><!-- -AOI# Marker- -></pre> <p>Расположение строки комментария определяет расположение вывода ODWEK. Строки, указанные до строки комментария, отображаются до вывода ODWEK. Строки, указанные после строки комментария, отображаются после вывода ODWEK.</p> <p>Файл <code>TEMPLATE.HTM</code> представляет собой пример файла шаблона, поставляемого вместе с программным обеспечением ODWEK. На основе примера файла шаблона вы можете создать собственный файл шаблона для функции входа в систему.</p>
<code>_frame= значение</code>	В вывод этой команды включается атрибут <code>target=значение</code> . Данный параметр упрощает процесс создания фреймов HTML. Это необязательный параметр.
<code>_datefmt= значение</code>	Определяет формат даты для поиска в базе данных и отображения элементов, соответствующих запросу. Формат даты по умолчанию устанавливается на странице Информация о поле папки. Более подробная информация о форматах даты, поддерживаемых OnDemand приведена в книге <i>Руководство администратора общего сервера IBM Content Manager OnDemand for iSeries V5R1</i> , SC27-1161.

Таблица 7. функция Вход в систему (продолжение)

Имя=Значение	Назначение
<code>_nohtml= значение</code>	Определяет тип вывода ODWEK. Значение по умолчанию - 0 (ноль). Оно означает, что создается вывод HTML. Если указано значение 1, то ODWEK создает вывод ASCII с ограничителями. Дополнительная информация о выводе ASCII с ограничителями в разделе Приложение H, “Не создавать вывод HTML”, на стр. 137.
<code>_port= значение</code>	Номер порта сервера OnDemand. Значение по умолчанию - 0 (ноль). Оно означает, что применяется порт, номер которого указан в Таблице служб (WRKSRVTBLE). Если номер порта не указан в Таблице служб, то по умолчанию применяется порт 1445. Указанное значение переопределяет значение параметра PORT из файла ARSWWW.INI.
<code>_codepage= значение</code>	Задаёт кодовую страницу базы данных OnDemand. По умолчанию применяется кодовая страница сервера HTTP. Кодовую страницу следует указывать, если кодовая страница сервера отличается от кодовой страницы базы данных. Указанное значение переопределяет значение параметра CODEPAGE из файла ARSWWW.INI.
<code>_cgibin= программа</code>	<p>Применяется программой CGI для создания следующей страницы вывода. Если этот параметр указан, то вместо программы по умолчанию (ARSWWW.CGI) в странице будет содержаться вызов к заданной программе. Как правило, этот параметр применяется программистами, разрабатывающими клиентские программы CGI или сервлеты для программ CGI или сервлетов, предоставленных IBM.</p> <p>В качестве <i>программы</i> можно указать каталог, относительно директивы ServerRoot, или <i>псевдоним</i>, определенный в файле конфигурации сервера HTTP. По умолчанию ODWEK загружает программу CGI из каталога CGI-BIN.</p>

Использование

Следующие параметры являются обязательными:

`_function`
`_server`
`_user`
`_password`

Следующие параметры являются необязательными:

`_new_password`
`_frame`
`_datefmt`
`_html`
`_nohtml`
`_port`
`_codepage`
`_logoff`
`_cgibin`

Пример вызова функции

```
http://www.company.com/cgi-bin/arswww.cgi?_function=logon
&_server=od400&_user=web&_password=web
&_html=template.htm
```

Печать документа (Сервер)

Отправляет документы на указанный принтер сервера

Цель

Функция Печать документа на сервере позволяет отправить копии документов на принтер сервера OnDemand. Для работы с функцией печати сервера пользователь должен обладать соответствующими правами доступа к группе приложений OnDemand. (Права доступа к группе приложений позволяют добавлять комментарии.) На указанном сервере OnDemand должен быть определен по крайней мере один принтер сервера.

Параметры

Таблица 8. Функция Печать документа

Имя=Значение	Назначение
_function=printdocs	Позволяет распечатать документ.
_server= значение	Имя сервера OnDemand.
_user= значение	ИД пользователя OnDemand. Для печати документов пользователь обладать правами доступа на печать документов для каждой группы приложений, в которых содержатся необходимые документы. (Права доступа к группе приложений позволяют распечатать документ.)
_password= значение	Пароль пользователя
_folder= значение	Имя папки.
_printer=значение	Имя принтера сервера OnDemand. Если указан Факс или Принтер с информацией, вы можете указать следующие дополнительные параметры: _recv_name=значение Имя получателя. _recv_comp=значение Название компании получателя. _recv_fax=значение Номер факса получателя. _send_name=значение Имя отправителя. _send_comp=значение Название компании отправителя. _send_tel=значение Номер телефона отправителя. _send_fax=значение Номер факса отправителя. _send_cover=значение Пользовательский модификатор, объединяемый программой выхода заголовка страницы с другими параметрами, для создания обложки документа. _subject=значение Тема документа. _notes=значение Заметки о документе.

Таблица 8. Функция Печать документа (продолжение)

Имя=Значение	Назначение
_html= значение	<p>Определяет файл HTML, применяемый ODWEK в качестве шаблона для создания Web-страниц вывода. В качестве значения можно указать имя файла или символ *. Если указана звездочка, то применяется файл PRINTDOCS.HTML, расположенный в каталоге, заданном с помощью параметра TEMPLATEDIR в файле ARSWWW.INI. Если указано имя файла без пути, то файл должен быть расположен в каталоге, заданном в параметре TEMPLATEDIR. Если вместе с именем файла указан путь к нему, он задается относительно каталога, указанного в параметре TEMPLATEDIR.</p> <p>Общий формат файла HTML определяется пользователем. Однако файл шаблона должен содержать следующую строку комментария:</p> <pre><!-- -AOI# Marker-- -></pre> <p>Расположение строки комментария определяет расположение вывода ODWEK. Строки, указанные до строки комментария, отображаются до вывода ODWEK. Строки, указанные после строки комментария, отображаются после вывода ODWEK.</p> <p>Файл TEMPLATE.HTM представляет собой пример файла шаблона, поставляемого вместе с программным обеспечением ODWEK. На основе примера файла шаблона вы можете создать собственный файл шаблона для функции печати документов.</p>
_nohtml= значение	<p>Определяет тип вывода ODWEK. Значение по умолчанию - 0 (ноль). Оно означает, что создается вывод HTML. Если указано значение 1, то ODWEK создает вывод ASCII с ограничителями. Дополнительная информация о выводе ASCII с ограничителями в разделе Приложение Н, "Не создавать вывод HTML", на стр. 137.</p>
_docids= список ИД документов	<p>Список ИД документов для печати. Идентификаторы документов возвращаются функцией Список найденных документов. Если указано несколько ИД документов, то они должны быть разделены с помощью символа \003.</p> <p>Примечание: Если число идентификаторов документов превышает 200, необходимо указать параметр _max_hits.</p>
_port= значение	<p>Номер порта сервера OnDemand. Значение по умолчанию - 0 (ноль). Оно означает, что применяется порт, номер которого указан в Таблице служб (WRKSRVTBLE). Если номер порта не указан в Таблице служб, то по умолчанию применяется порт 1445. Указанное значение переопределяет значение параметра PORT из файла ARSWWW.INI.</p>
_codepage= значение	<p>Задаёт кодовую страницу базы данных OnDemand. По умолчанию применяется кодовая страница сервера HTTP. Кодовую страницу следует указывать, если кодовая страница сервера отличается от кодовой страницы базы данных. Указанное значение переопределяет значение параметра CODEPAGE из файла ARSWWW.INI.</p>

Таблица 8. Функция Печать документа (продолжение)

Имя=Значение	Назначение
<code>_max_hits=значение</code>	<p>Этот параметр позволяет указать число идентификаторов документов для обработки. Указанное значение должно быть больше числа идентификаторов документов, которое указано в параметре <code>_docids</code>, или равно ему.</p> <p>Примечание: Если число идентификаторов документов превышает значение, указанной в параметре <code>MAXHITS</code> файла <code>ARSWWW.CGI</code> (или 200, если значение не указано), необходимо указать параметр <code>_max_hits</code>. Если параметр <code>_max_hits</code> не указан (или не указано значение параметра <code>MAXHITS</code>), то обрабатывается максимальное число идентификаторов документов, равное 200. При этом значение параметра <code>_docids</code> не учитывается.</p> <p>Для определения числа обрабатываемых документов применяются следующие значения:</p> <ul style="list-style-type: none"> • Значение параметра <code>_max_hits</code>, если он указан. Это значение переопределяет значение параметра <code>MAXHITS</code>. • Значение параметра <code>MAXHITS</code>, если оно указано. • Если не указано ни одно из приведенных выше значений, то по умолчанию обрабатывается максимальное число идентификаторов документов, равное 200.
<code>_logoff=1</code>	<p>Автоматически отключает пользователя от сервера после печати документа. Данный параметр позволяет избежать необходимости вызова функции Выход из системы для отключения пользователя. Для этого параметра допустимо только значение 1.</p>

Использование

Следующие параметры являются обязательными:

```

_function
_server
_user
_password
_folder
_printer
_docids

```

Следующие параметры являются необязательными:

```

_recv_name
_recv_comp
_recv_fax
_send_name
_send_comp
_send_tel
_send_fax
_send_cover
_subject
_notes
_max_hits
_html
_nohtml
_port
_codepage
_logoff

```

Пример вызова функции

```
http://www.company.com/cgi-bin/arswww.cgi?_function=printdocs  
&_server=od400&_user=web&_password=web  
&_folder=credit%20card%20statements  
&_printer=infoprint60  
&_docids=6850-6851-SUA17-1FAAA-225712-1634-132014-132172-89-76-11-25-0  
&_logoff=1
```

Загрузить документ

Загружает выбранный документ с сервера OnDemand

Цель

Функция Загрузить документ позволяет загрузить документ с сервера OnDemand. Документ отображается в окне браузера с помощью аплета, программы просмотра или другой программы, связанной с документами этого типа.

Параметры

Таблица 9. Функция Загрузить документ

Имя=значение	Назначение
<code>_function=retrieve</code>	Позволяет загрузить выбранный документ.
<code>_server= значение</code>	Имя сервера OnDemand.
<code>_user= значение</code>	ИД пользователя OnDemand.
<code>_password= значение</code>	Пароль для ИД пользователя.
<code>_folder= значение</code>	Имя папки.
<code>folder field name=значение</code>	Имя поля поиска папки и значение поиска. Вы можете указать несколько наборов имен полей и значений поиска, не превышающих число полей, определенных для папки.
<code>_html=значение</code>	<p>Определяет файл HTML, используемый в качестве шаблона для создания Web-страницы, отображаемый в случае ошибки при загрузке документа. В качестве значения можно указать имя файла или символ *. Если указана звездочка, то применяется файл RETRIEVE.HTML, расположенный в каталоге, заданном с помощью параметра TEMPLATEDIR в файле ARSWWW.INI. Если указано имя файла без пути, то файл должен быть расположен в каталоге, заданном в параметре TEMPLATEDIR. Если вместе с именем файла указан путь к нему, он задается относительно каталога, указанного в параметре TEMPLATEDIR.</p> <p>Общий формат файла HTML определяется пользователем. Однако файл шаблона должен содержать следующую строку комментария:</p> <pre><!-- - -AOI# Marker-- - -></pre> <p>Расположение строки комментария определяет расположение вывода ODWEK. Строки, указанные до строки комментария, отображаются до вывода ODWEK. Строки, указанные после строки комментария, отображаются после вывода ODWEK.</p> <p>Файл TEMPLATE.HTM представляет собой пример файла шаблона, поставляемого вместе с программным обеспечением ODWEK. На основе примера файла шаблона вы можете создать собственный файл шаблона для функции загрузки документов.</p>
<code>_nohtml= значение</code>	Определяет тип вывода ODWEK. Значение по умолчанию - 0 (ноль). Оно означает, что создается вывод HTML. Если указано значение 1, то ODWEK создает вывод ASCII с ограничителями. Дополнительная информация о выводе ASCII с ограничителями в разделе Приложение Н, "Не создавать вывод HTML", на стр. 137.
<code>_port= значение</code>	Номер порта сервера OnDemand. Значение по умолчанию - 0 (ноль). Оно означает, что применяется порт, номер которого указан в Таблице служб (WRKSRVTBLE). Если номер порта не указан в Таблице служб, то по умолчанию применяется порт 1445. Указанное значение переопределяет значение параметра PORT из файла ARSWWW.INI.

Таблица 9. Функция Загрузить документ (продолжение)

Имя=Значение	Назначение
_codepage=значение	<p>Задает кодовую страницу базы данных OnDemand. По умолчанию применяется кодовая страница сервера HTTP. Кодовую страницу следует указывать, если кодовая страница сервера отличается от кодовой страницы базы данных. Указанное значение переопределяет значение параметра CODEPAGE из файла ARSWWW.INI.</p>
_cgibin=программа	<p>Применяется программой CGI для создания следующей страницы вывода. Если этот параметр указан, то вместо программы по умолчанию (ARSWWW.CGI) в странице будет содержаться вызов к заданной программе. Как правило этот параметр применяется программистами, разрабатывающими клиентские программы CGI или сервлеты для программ CGI или сервлетов, предоставленных IBM.</p> <p>В качестве <i>программы</i> можно указать каталог, относительно директивы ServerRoot, или <i>псевдоним</i>, определенный в файле конфигурации сервера HTTP. По умолчанию ODWEK загружает программу CGI из каталога CGI-BIN.</p>
_or=значение	<p>Если указано значение 1, то поля поиска объединяются с помощью логического оператора ИЛИ. При этом документ должен соответствовать по крайней мере одному из указанных значений поиска. Значение по умолчанию - 0 (ноль). Оно означает, что поля поиска объединяются с помощью оператора И (документ должен соответствовать всем указанным значениям поиска).</p>
_afp=значение	<p>Данное значение определяет действие, выполняемое в процессе загрузки документа AFP с сервера OnDemand, перед отправкой документа клиенту. Например, можно преобразовать документы AFP в формат HTML с помощью программы AFP2WEB Transform и отобразить вывод HTML с помощью апплета AFP2HTML. Для этого следует указать параметр _afp=HTML чтобы перед отправкой клиенту документы AFP преобразовывались.</p> <p>Предусмотрены следующие значения:</p> <p>ASCII Документы AFP преобразуются в текст ASCII.</p> <p>HTML Документы AFP преобразуются в документы HTML с помощью программы AFP2WEB Transform.</p> <p>NATIVE Документы и ресурсы AFP загружаются с сервера OnDemand и распаковываются. Примечание: Если указан параметр _afp=NATIVE, убедитесь, что для типа данных MIME определена необходимая программа просмотра (дополнительная информация приведена в разделе “[MIMETYPES]” на стр. 23).</p> <p>PDF Документы AFP преобразуются в документы PDF с помощью программы AFP2WEB Transform.</p> <p>PLUGIN Документы AFP не преобразуются (по умолчанию).</p>
_email=значение	<p>Данное значение определяет действие, выполняемое в процессе загрузки документа EMAIL с сервера OnDemand, перед отправкой документа клиенту. Предусмотрены следующие значения:</p> <p>NATIVE Документы EMAIL загружаются с сервера OnDemand и распаковываются. Примечание: Если указан параметр _email=NATIVE, убедитесь, что для типа данных MIME определена необходимая программа просмотра (дополнительная информация приведена в разделе “[MIMETYPES]” на стр. 23).</p> <p>HTML Документы EMAIL преобразуются в формат HTML.</p>

Таблица 9. Функция Загрузить документ (продолжение)

Имя=Значение	Назначение
_line=значение	<p>Данное значение определяет действие, выполняемое в процессе загрузки документа строковых данных с сервера OnDemand, перед отправкой документа клиенту. Предусмотрены следующие значения:</p> <p>APPLET Документы строковых данных преобразуются для просмотра с помощью апплета Строковые данные (по умолчанию).</p> <p>ASCII Документы строковых данных преобразуются в формат ASCII.</p> <p>NATIVE Документы строковых данных загружаются с сервера OnDemand и распаковываются. Примечание: Если указан параметр <code>_line=NATIVE</code>, убедитесь, что для типа данных MIME определена необходимая программа просмотра (дополнительная информация приведена в разделе “[MIMETYPES]” на стр. 23).</p>
_docid= ИД документа	Идентификатор загружаемого документа. Идентификатор документа возвращается функцией Список найденных документов.
_logoff=1	Автоматически отключает пользователя от сервера после загрузки документа. Данный параметр позволяет избежать необходимости вызова функции Выход из системы для отключения пользователя. Для этого параметра допустимо только значение 1.

Использование

Следующие параметры являются обязательными:

```
_function
_server
_user
_password
_folder
```

Следующие параметры являются необязательными:

```
folder field name
_docid
_or
_afp
_email
_line
_html
_nohtml
_port
_codepage
_logoff
_cgibin
```

Пример вызова функции

```
http://www.company.com/cgi-bin/arswww.cgi?_function=retrieve
&_server=od400&_user=web&_password=web
&_folder=credit%20card%20statements
&account%20number=1000100010009999&date=1%2f1%2f96
&_html=template.htm&_logoff=1
```

Критерии поиска

Отображает критерии поиска, заданные для выбранной папки

Цель

Функция Критерии поиска отображает позволяет просмотреть критерии поиска для указанной папки с помощью формы. Пользователь может выбрать критерии поиска по умолчанию или ввести критерии для поиска конкретных документов. При нажатии кнопки Отправить открывается Web-страница, в которой приведен список документов, соответствующих критериям поиска.

Параметры

Таблица 10. Функция Критерии поиска

Имя=значение	Назначение
_function=критерии поиска	Отображает критерии поиска, заданные для выбранной папки.
_server=значение	Имя сервера OnDemand.
_user=значение	ИД пользователя OnDemand.
_password=значение	Пароль для ИД пользователя.
_folder=значение	Имя папки.
_html=значение	<p>Определяет файл HTML, применяемый ODWEK в качестве шаблона для создания Web-страниц вывода. В качестве значения можно указать имя файла или символ *. Если указана звездочка, то применяется файл SEARCHCRIT.HTML, расположенный в каталоге, заданном с помощью параметра TEMPLATEDIR в файле ARSWWW.INI. Если указано имя файла без пути, то файл должен быть расположен в каталоге, заданном в параметре TEMPLATEDIR. Если вместе с именем файла указан путь к нему, он задается относительно каталога, указанного в параметре TEMPLATEDIR.</p> <p>Общий формат файла HTML определяется пользователем. Однако файл шаблона должен содержать следующую строку комментария:</p> <pre><!-- -AOI# Marker-- -></pre> <p>Расположение строки комментария определяет расположение вывода ODWEK. Строки, указанные до строки комментария, отображаются до вывода ODWEK. Строки, указанные после строки комментария, отображаются после вывода ODWEK.</p> <p>Файл TEMPLATE.HTM представляет собой пример файла шаблона, поставляемого вместе с программным обеспечением ODWEK. На основе примера файла шаблона вы можете создать собственный файл шаблона для функции критериев поиска.</p>
_frame=значение	В вывод этой команды включается атрибут target=значение. Данный параметр упрощает процесс создания фреймов HTML. Это необязательный параметр.
_datefmt=значение	Определяет формат даты для поиска в базе данных и отображения элементов, соответствующих запросу. Формат даты по умолчанию устанавливается на странице Информация о поле папки. Более подробная информация о форматах даты, поддерживаемых OnDemand приведена в книге <i>Руководство администратора общего сервера IBM Content Manager OnDemand for iSeries V5R1</i> , SC27-1161.

Таблица 10. Функция Критерии поиска (продолжение)

Имя=Значение	Назначение
<code>_nohtml= значение</code>	Определяет тип вывода ODWEK. Значение по умолчанию - 0 (ноль). Оно означает, что создается вывод HTML. Если указано значение 1, то ODWEK создает вывод ASCII с ограничителями. Дополнительная информация о выводе ASCII с ограничителями в разделе Приложение Н, "Не создавать вывод HTML", на стр. 137.
<code>_port=значение</code>	Номер порта сервера OnDemand. Значение по умолчанию - 0 (ноль). Оно означает, что применяется порт, номер которого указан в Таблице служб (WRKSRVTBLE). Если номер порта не указан в Таблице служб, то по умолчанию применяется порт 1445. Указанное значение переопределяет значение параметра PORT из файла ARSWWW.INI.
<code>_codepage=значение</code>	Задает кодовую страницу базы данных OnDemand. По умолчанию применяется кодовая страница сервера HTTP. Кодовую страницу следует указывать, если кодовая страница сервера отличается от кодовой страницы базы данных. Указанное значение переопределяет значение параметра CODEPAGE из файла ARSWWW.INI.
<code>_cgibin=программа</code>	<p>Применяется программой CGI для создания следующей страницы вывода. Если этот параметр указан, то вместо программы по умолчанию (ARSWWW.CGI) в странице будет содержаться вызов к заданной программе. Как правило этот параметр применяется программистами, разрабатывающими клиентские программы CGI или сервлеты для программ CGI или сервлетов, предоставленных IBM.</p> <p>В качестве <i>программы</i> можно указать каталог, относительно директивы ServerRoot, или <i>псевдоним</i>, определенный в файле конфигурации сервера HTTP. По умолчанию ODWEK загружает программу CGI из каталога CGI-BIN.</p>
<code>_logoff=1</code>	Автоматически отключает пользователя от сервера после отображения критериев поиска. Данный параметр позволяет избежать необходимости вызова функции Выход из системы для отключения пользователя. Для этого параметра допустимо только значение 1.

Использование

Следующие параметры являются обязательными:

`_function`
`_server`
`_user`
`_password`
`_folder`

Следующие параметры являются необязательными:

`_frame`
`_datefmt`
`_html`
`_nohtml`
`_port`
`_codepage`
`_logoff`
`_cgibin`

Пример вызова функции

```
http://www.company.com/cgi-bin/arswww.cgi?_function=searchcrit  
&_server=od400&_user=web&_password=web  
&_folder=credit%20card%20statements&_html=template.htm  
&_logoff=1
```

|

Обновить документ

Обновляет значения базы данных для указанного документа.

Цель

Функция Обновить документ позволяет пользователям, обладающим соответствующими правами доступа, обновить документы. Функция Обновить документ обновляет одно или несколько значений базы данных указанного документа.

Параметры

Таблица 11. функция Обновить документ

Имя=Значение	Назначение
<code>_function=updatedoc</code>	Обновляет базу данных.
<code>_server=значение</code>	Имя сервера OnDemand.
<code>_user=значение</code>	ИД пользователя OnDemand. Пользователь должен обладать правами доступа на Обновление документов для группы приложений.
<code>_password=значение</code>	Пароль пользователя
<code>_folder=значение</code>	Имя папки.
<code>folder field name=значение</code>	Задаёт имя обновляемого поля, а также новое значение. Вы можете указать несколько наборов имен полей и значений. Их число не должно превышать число полей, определенных для папки.
<code>_html=значение</code>	<p>Определяет файл HTML, применяемый ODWEK в качестве шаблона для создания Web-страниц вывода. В качестве значения можно указать имя файла или символ *. Если указана звездочка, то применяется файл UPDATE.HTML, расположенный в каталоге, заданном с помощью параметра <code>TEMPLATEDIR</code> в файле <code>ARSWWW.INI</code>. Если указано имя файла без пути, то файл должен быть расположен в каталоге, заданном в параметре <code>TEMPLATEDIR</code>. Если вместе с именем файла указан путь к нему, он задается относительно каталога, указанного в параметре <code>TEMPLATEDIR</code>.</p> <p>Общий формат файла HTML определяется пользователем. Однако файл шаблона должен содержать следующую строку комментария:</p> <pre><!-- - -AOI# Marker-- - -></pre> <p>Расположение строки комментария определяет расположение вывода ODWEK. Строки, указанные до строки комментария, отображаются до вывода ODWEK. Строки, указанные после строки комментария, отображаются после вывода ODWEK.</p> <p>Файл <code>TEMPLATE.HTM</code> представляет собой пример файла шаблона, поставляемого вместе с программным обеспечением ODWEK. На основе примера файла шаблона вы можете создать собственный файл шаблона для функции обновления.</p>
<code>_nohtml=значение</code>	Определяет тип вывода ODWEK. Значение по умолчанию - 0 (ноль). Оно означает, что создается вывод HTML. Если указано значение 1, то ODWEK создает вывод ASCII с ограничителями. Дополнительная информация о выводе ASCII с ограничителями в разделе Приложение Н, "Не создавать вывод HTML", на стр. 137.
<code>_docid= ИД документа</code>	Идентификатор обновляемого документа. Идентификатор документа возвращается функцией Список найденных документов.

Таблица 11. функция Обновить документ (продолжение)

Имя=Значение	Назначение
_port=значение	Номер порта сервера OnDemand. Значение по умолчанию - 0 (ноль). Оно означает, что применяется порт, номер которого указан в Таблице служб (WRKSRVTBLE). Если номер порта не указан в Таблице служб, то по умолчанию применяется порт 1445. Указанное значение переопределяет значение параметра PORT из файла ARSWWW.INI.
_codepage=значение	Задаёт кодовую страницу базы данных OnDemand. По умолчанию применяется кодовая страница сервера HTTP. Кодовую страницу следует указывать, если кодовая страница сервера отличается от кодовой страницы базы данных. Указанное значение переопределяет значение параметра CODEPAGE из файла ARSWWW.INI.
_logoff=1	Автоматически отключает пользователя от сервера после обновления документа. Данный параметр позволяет избежать необходимости вызова функции Выход из системы для отключения пользователя. Для этого параметра допустимо только значение 1.

Использование

Следующие параметры являются обязательными:

_function
 _server
 _user
 _password
 _folder

Следующие параметры являются необязательными:

folder field name
 _docid
 _html
 _nohtml
 _port
 _codepage
 _logoff

Пример вызова функции

```

http://www.company.com/cgi-bin/arswww.cgi?_function=updatedoc
&_server=od400&_user=web&_password=web
&_folder=credit%20card%20statements
&account%20number=1000100010009999
&_docid=6850-6851-SUA17-1FAAA-225712-1634-132014-132172-89-76-11-25-0
&_html=template.htm&_logoff=1
  
```

Просмотреть комментарии

Просмотр комментариев, вложенных в указанный документ

Цель

Функция Просмотреть комментарии позволяет просмотреть комментарии, добавленные к указанному документу. Для того чтобы просмотреть комментарии пользователь должен обладать соответствующими правами доступа к группе приложений OnDemand. (Права доступа к группе приложений позволяют добавлять комментарии.)

Параметры

Таблица 12. функция Просмотреть комментарии

Имя=Значение	Назначение
<code>_function=getnotes</code>	Просмотр комментариев
<code>_server=значение</code>	Имя сервера OnDemand.
<code>_user=значение</code>	ИД пользователя OnDemand. Для того чтобы просмотреть комментарии пользователь должен обладать правами доступа на просмотр комментариев для каждой группы приложений, в которых содержатся необходимые документы. (Права доступа к группе приложений позволяют просматривать комментарии.)
<code>_password=значение</code>	Пароль пользователя
<code>_folder=значение</code>	Имя папки.
<code>_html=значение</code>	<p>Определяет файл HTML, применяемый ODWEK в качестве шаблона для создания Web-страниц вывода. В качестве значения можно указать имя файла или символ *. Если указана звездочка, то применяется файл GETNOTES.HTML, расположенный в каталоге, заданном с помощью параметра TEMPLATEDIR в файле ARSWWW.INI. Если указано имя файла без пути, то файл должен быть расположен в каталоге, заданном в параметре TEMPLATEDIR. Если вместе с именем файла указан путь к нему, он задается относительно каталога, указанного в параметре TEMPLATEDIR.</p> <p>Общий формат файла HTML определяется пользователем. Однако файл шаблона должен содержать следующую строку комментария:</p> <pre><!-- -AOI# Marker-- -></pre> <p>Расположение строки комментария определяет расположение вывода ODWEK. Строки, указанные до строки комментария, отображаются до вывода ODWEK. Строки, указанные после строки комментария, отображаются после вывода ODWEK.</p> <p>Файл TEMPLATE.HTM представляет собой пример файла шаблона, поставляемого вместе с программным обеспечением ODWEK. На основе примера файла шаблона вы можете создать собственный файл шаблона для функции просмотра комментариев.</p>
<code>_nohtml=значение</code>	Определяет тип вывода ODWEK. Значение по умолчанию - 0 (ноль). Оно означает, что создается вывод HTML. Если указано значение 1, то ODWEK создает вывод ASCII с ограничителями. Дополнительная информация о выводе ASCII с ограничителями в разделе Приложение Н, "Не создавать вывод HTML", на стр. 137.
<code>_docid=ИД документа</code>	Идентификатор документа, содержащего комментарии для просмотра. Идентификатор документа возвращается функцией Список найденных документов.

Таблица 12. функция Просмотреть комментарии (продолжение)

Имя=Значение	Назначение
<code>_port=значение</code>	Номер порта сервера OnDemand. Значение по умолчанию - 0 (ноль). Оно означает, что применяется порт, номер которого указан в Таблице служб (WRKSRVTBLE). Если номер порта не указан в Таблице служб, то по умолчанию применяется порт 1445. Указанное значение переопределяет значение параметра PORT из файла ARSWWW.INI.
<code>_codepage=значение</code>	Задаёт кодовую страницу базы данных OnDemand. По умолчанию применяется кодовая страница сервера HTTP. Кодовую страницу следует указывать, если кодовая страница сервера отличается от кодовой страницы базы данных. Указанное значение переопределяет значение параметра CODEPAGE из файла ARSWWW.INI.
<code>_logoff=1</code>	Автоматически отключает пользователя от сервера после просмотра комментария. Данный параметр позволяет избежать необходимости вызова функции Выход из системы для отключения пользователя. Для этого параметра допустимо только значение 1.

Использование

Следующие параметры являются обязательными:

```
_function
_server
_user
_password
_folder
_docid
```

Следующие параметры являются необязательными:

```
_html
_nohtml
_port
_codepage
_logoff
```

Пример вызова функции

```
http://www.company.com/cgi-bin/arswww.cgi?_function=getnotes
&_server=od400&_user=web&_password=web
&_folder=credit%20card%20statements
&_docid=6850-6851-SUA17-1FAAA-225712-1634-132014-132172-89-76-11-25-0
&_logoff=1
```

Приложение В. Справочник по сервлетам Java

Сервлет Java работает в качестве контроллера Web-приложения и выполняет функции и общие задачи до и после действия, такие как управление соединением с сервером OnDemand.

Для типичных заданий приложений предусмотрены следующие функции:

- вход в систему и выход из нее
- поиск
- загрузка, печать и обновление документов
- добавление и просмотр комментариев
- изменение пароля

Для применения сервлета в приложении используется набор функций приложений и параметров.

Сервлет Java работает с теми же функциями, что и программа CGI. Справка по функциям, а также их описание и параметры приведены в разделе Приложение А, “Справочник по API CGI”, на стр. 59.

Приложение С. Справочник по API Java

Документация по API Java поставляется вместе с программным обеспечением ODWEK в формате HTML.

Для просмотра документации следует установить программное обеспечение ODWEK и извлечь файлы документации из файла ODApiDoc.zip, расположенного в каталоге /QIBM/ProdData/OnDemand/www/. В процессе распаковки для файлов архива должна сохраняться структура каталогов.

Для просмотра документации после извлечения соответствующих файлов откройте файл index.html с помощью Web-браузера.

Приложение D. Руководство программиста API Java

Прикладные программные интерфейсы Java (API) представляют собой набор классов, предназначенных для обращения к данным сервера OnDemand и управления ими. В этом разделе приведено описание API Java, реализации функций работы с документами с помощью Java, а также подключения к Internet.

API Java поддерживают следующие функции:

- Общая модель объектов для обращения к данным
- Поиск и обновление данных на серверах OnDemand. **Примечание:** В разделе Глава 1, “Обзор”, на стр. 1 приведены ограничения, применяемые при работе с сервером OnDemand for OS/390 версии 2.
- Реализация технологии клиент-сервер для пользователей приложений Java

Архитектура клиент-сервер

Данные API предоставляют удобный программный интерфейс для пользователей приложений. API могут быть расположены как на сервере OnDemand, так и на клиенте (в обоих случаях предусмотрен одинаковый интерфейс). Приложения могут быть расположены локально и удаленно. API клиента обращается к серверу для получения доступа к данным по сети. Обмен данными между клиентом и сервером выполняется с помощью классов. Дополнительные программы добавлять необязательно.

Классы API входят в состав одного пакета: `com.ibm.edms.od`.

Структура среды Java

Классы API входят в состав одного пакета: `com.ibm.edms.od`. В этом пакете предусмотрены следующие классы:

`com.ibm.edms.od.ODCallback`

Этот класс применяется во всех методах, в которых в процессе обработки операция сервера возвращает данные.

`com.ibm.edms.od.ODCriteria`

Этот класс представляет критерии поиска папки OnDemand. Класс критериев содержит методы, позволяющие указать оператор поиска, а также значения поиска.

`com.ibm.edms.od.ODException`

Этот класс представляет исключительные ситуации, которые могут возникнуть при работе с API.

`com.ibm.edms.od.ODFolder`

Этот класс представляет папку OnDemand. Данный объект возвращается в случае успешного вызова функции `ODServer.openFolder()`. Кроме того, в этом классе содержится информация о критериях папки. Для сужения поиска на сервере в эти объекты следует внести изменения.

`com.ibm.edms.od.ODHit`

Этот класс представляет документ OnDemand.

`com.ibm.edms.od.ODNote`

Этот класс представляет комментарий OnDemand.

com.ibm.edms.od.ODServer

Этот класс представляет соединение с сервером OnDemand. С помощью этого класса вы можете войти в систему, выйти из нее, а также изменить пароль. После успешного входа в систему в этом объекте будет содержаться список всех папок, доступных в сеансе. **Примечание:** К этому объекту сервера следует обращаться с помощью среды без нитей. Единственным исключением в этом случае является отмена операции сервера.

Советы программисту

Пакет `com.ibm.edms.od` необходимо импортировать в приложение ODWEK.

Для выполнения приложений ODWEK, применяющих API Java, не требуется сервер HTTP или Web-сервер приложений. Приложения ODWEK можно запускать с помощью интерпретатора Java.

Для запуска приложения ODWEK с помощью интерпретатора Java выполните следующие действия:

1. Скопируйте файл `arswww.ini` в пользовательский каталог выполнения.
2. Скопируйте общую библиотеку в каталог, в который был скопирован файл `arswww.ini`:

Таблица 13. Имя файла общей библиотеки

Операционная система	Общая библиотека
AIX	libarswwwsl.a
HP-UX	libarswwwsl.sl
Linux	libarswwwsl.so
Solaris	libarswwwsl.so
Windows	arswwwsl.dll

3. При работе с операционными системами Windows скопируйте следующие файлы в каталог, в который был скопирован файл `arswww.ini`:
ARSSCKNT.DLL
ARSCT32.DLL
4. При запуске приложения с помощью интерпретатора Java укажите имя пользовательского каталога. Пример приведен в разделе “Выполнение приложения ODWEK” на стр. 95.

Настройка среды системы

В процессе настройки среды AIX, HP-UX, Linux, Solaris или Windows следует указать следующие параметры:

package

Импорт для всех приложений ODWEK.

- `com.ibm.edms.od`

Файлы библиотеки

Общие объекты для операционных систем AIX, HP-UX, Linux и Solaris

Библиотеки DLL для Windows

Установка переменных среды

В процессе разработки приложения ODWEK необходимо установить пользовательскую среду.

AIX

В среде AIX для установки среды разработки приложений ODWEK следует задать следующие переменные среды.

PATH Для переменной PATH должно быть указано значение
/usr/lpp/ars/www

LIBPATH Для переменной LIBPATH должно быть указано значение
/usr/lpp/ars/www

LD_LIBRARY_PATH
Для переменной LD_LIBRARY_PATH должно быть указано значение
/usr/lpp/ars/www

CLASSPATH Для переменной CLASSPATH должна быть указана библиотека класса
/usr/lpp/ars/www/api/ODApi.jar.

HP-UX

В среде HP-UX для установки среды разработки приложений ODWEK следует задать следующие переменные среды.

PATH Для переменной PATH должно быть указано значение
/opt/ondemand/www

LIBPATH Для переменной LIBPATH должно быть указано значение
/opt/ondemand/www

LD_LIBRARY_PATH
Для переменной LD_LIBRARY_PATH должно быть указано значение
/opt/ondemand/www

CLASSPATH Для переменной CLASSPATH должна быть указана библиотека класса
/opt/ondemand/www/api/ODApi.jar.

Linux

В среде Linux для установки среды разработки приложений ODWEK следует задать следующие переменные среды.

PATH Для переменной PATH должно быть указано значение
/opt/ondemand/www

LIBPATH Для переменной LIBPATH должно быть указано значение
/opt/ondemand/www

LD_LIBRARY_PATH
Для переменной LD_LIBRARY_PATH должно быть указано значение
/opt/ondemand/www

CLASSPATH Для переменной CLASSPATH должна быть указана библиотека класса
/opt/ondemand/www/api/ODApi.jar.

Solaris

В среде Solaris для установки среды разработки приложений ODWEK следует задать следующие переменные среды.

PATH Для переменной PATH должно быть указано значение
/opt/ondemand/www

LIBPATH Для переменной LIBPATH должно быть указано значение /opt/ondemand/www

LD_LIBRARY_PATH Для переменной LD_LIBRARY_PATH должно быть указано значение /opt/ondemand/www

CLASSPATH Для переменной CLASSPATH должна быть указана библиотека класса /opt/ondemand/www/api/ODApi.jar.

Windows

В среде Windows для установки среды разработки приложений ODWEK следует задать следующие переменные среды.

PATH Для переменной PATH должно быть указано значение `x:\uuuuuuu\DLL`, где `x` - это диск, на котором установлена программа ODWEK, а `uuuuuuu` - это установочный каталог программного обеспечения ODWEK.

CLASSPATH Для переменной CLASSPATH должно быть указано значение `x:\uuuuuuu\WWW\API\ODApi.jar`, где `x` - это диск, на котором установлена программа ODWEK, а `uuuuuuu` - это установочный каталог библиотеки класса.

Трассировка и информация диагностики

Для обработки неполадок, возникающих в приложениях API Java рекомендуется применять средства трассировки и обработки исключительных ситуаций.

Трассировка

Следующие параметры, указанные в файле ARSWWW.INI, позволяют записывать информацию трассировки в файл `arswww.log`, расположенный в заданном каталоге:

```
[DEBUG]
LOG=1
LOGDIR=/ars/www/log
```

Примечание: IBM рекомендует разрешать ведение протокола только в тех случаях, когда это необходимо, например, для воспроизведения неполадки, так как при этом в файл протокола записываются значительные объемы информации. Если вам необходимо разрешить ведение протокола на длительный период времени, убедитесь, что файл протокола расположен на устройстве, в котором доступно достаточно памяти. Рекомендуется периодически удалять старые файлы протоколов.

Дополнительная информация о других инструментах, позволяющих собирать информацию о системе и документах, приведена в разделе Приложение J, “Инструменты определения неполадок”, на стр. 143.

Обработка исключительных ситуаций

API Java выбрасывают исключительные ситуации в случае обнаружения неполадок. Выбрасывание исключения приводит к созданию объекта исключения из класса `ODException` или одного из его подклассов.

При создании объекта `ODException` API заносит в файл протокола диагностическую информацию, если ведение протокола разрешено. Дополнительная информация о файле протокола, применяемого API Java, приведена в разделе “Трассировка”.

При получении `ODException` вы можете просмотреть сообщения об ошибках, коды ошибок, а также состояния ошибок, произошедших при выполнении приложения. В случае получения ошибки выдается сообщение об ошибке, содержащее расположение, в котором было выброшено исключение. Кроме того, указываются ИД ошибки и ИД исключения. Исходный код, приведенный ниже, является примером процесса выбрасывания и получения:

```
try
{
    odServer = new ODServer( );
    odServer.initialize( argv[9], "TcUpdate.java" );
    System.out.println( "Logging on to " + argv[0] + "..." );
    odServer.logon( argv[0], argv[1], argv[2] );
    odServer.logoff( );
    odServer.terminate( );
}

catch ( ODException e )
{
    System.out.println( "ODException: " + e );
    System.out.println( "    id = " + e.getErrorId( ) );
    System.out.println( "    msg = " + e.getErrorMsg( ) );
    e.printStackTrace( );
}
```

Константы

Описание констант, применяемых в API Java, приведено в электронном руководстве. За дополнительной информацией обратитесь к разделу Приложение С, “Справочник по API Java”, на стр. 89.

Выполнение приложения ODWEK

Интерпретатор Java позволяет выполнять приложения ODWEK. При создании, компиляции и выполнении приложений ODWEK обратите внимание на следующие особенности:

1. Создавайте приложения ODWEK с помощью методов, доступных в API Java. Импортируйте пакет API Java в файл приложения ODWEK. Например:

```
//*****
import java.util.*;
import java.io.*;
import com.ibm.edms.od.*;

public class Logon
{
    public static void main ( String argv[] )
    {
        .
        .
        .
    }
}
```

2. Компилируйте файл приложения ODWEK (`.java`) вместе с `javac` для создания файла `.class`. За инструкциями по компиляции приложений Java обратитесь к справочной публикации Java.
3. Запустите интерпретатор Java в приложении (`.class file`). Например:
`java Logon сервер ИД-пользователя пароль /tmp/ondemand/www`

Где `Logon` - это имя файла `.class`, `сервер`, `ИД-пользователя` и `пароль` - параметры приложения, а `/tmp/ondemand/www` - пользовательский каталог выполнения, в

котором содержится копия файла `arswww.ini`. **Примечание:** В этом примере предполагается, что путь к каталогам класса ODWEK и библиотек сервлета указан с помощью переменных среды системы (см. “Настройка среды системы” на стр. 92).

Подключение к серверу OnDemand

Объект класса `ODServer` представляет соединение с сервером OnDemand, позволяет управлять им, а также обеспечивает поддержку транзакций и выполняет команды сервера. В разделе Приложение С, “Справочник по API Java”, на стр. 89 приведена информация о том, где найти электронные справочники по методам и их описания.

При подключении к серверу OnDemand следует учитывать требования сервера. Например, длина пароля сервера OnDemand не может превышать восемь символов.

Установка соединения

В классе `ODServer` предусмотрены методы, необходимые для подключения к серверу OnDemand и отключения от него. В следующем примере применяется сервер библиотеки OnDemand `LIBSRVR1`, ИД пользователя `ADMIN` и пароль `PASSWD`. В примере для сервера OnDemand создается объект `ODServer`, устанавливается соединение с сервером, выполняются необходимые действия (не указаны в примере), а затем соединение прерывается.

```
odServer = new ODServer( );
odServer.initialize( "c:\odwekdir", "Пример" );
System.out.println( "Logging on to " + "LIBSRVR1" + "..." );
odServer.logon( "LIBSRVR1", "ADMIN", "PASSWD" );
.
.
odServer.logoff( );
odServer.terminate( );
```

В разделе “Работа с сервером OnDemand” на стр. 97 приведен полный пример приложения, из которого взят этот фрагмент.

Установка и просмотр паролей

Методы `ODServer` позволяют задавать и просматривать пароли сервера OnDemand. В следующем примере показано, каким образом можно задать и просмотреть пароль пользователя сервера библиотеки OnDemand.

```
odServer = new ODServer( );
odServer.setServer( "LIBSRVR1" );
odServer.setUserId( "ADMIN" );
odServer.setPassword( "PASSWD" );

System.out.println( "Logging on to " + "LIBSRVR1" + "..." );

odServer.logon( odServer.getServerName( ),
                odServer.getUserId( ),
                odServer.getPassword( ),
                ODConstant.CONNECT_TYPE_LOCAL,
                0 );
```

В разделе “Работа с сервером OnDemand” на стр. 97 приведен полный пример приложения, из которого взят этот фрагмент.

Работа с сервером OnDemand

Объект класса `ODServer` представляет соединение с сервером OnDemand, позволяет управлять им, а также обеспечивает поддержку транзакций и выполняет команды сервера.

В следующем примере с помощью методов `ODServer` подготавливается вход в систему, задается имя приложения, (при необходимости) отображается локальный каталог, отображается имя сервера, ИД пользователя, отображается и задается тип соединения и номер порта, а также выполняется отключение от сервера.

В данном примере применяются следующие методы `ODServer`:

- `initialize`
- `logon`
- `logoff`
- `terminate`
- `getConnectType`
- `getLocalDir`
- `getPassword`
- `getPort`
- `getServerName`
- `getUserId`
- `setApplicationName`
- `setConnectType`
- `setLocalDir`
- `setPassword`
- `setPort`
- `setServer`
- `setUserId`

Кроме того, применяются следующие параметры выполнения:

- Имя сервера
- ИД пользователя
- Пароль
- Каталог конфигурации (расположение файла `arswww.ini`)
- (дополнительно) Локальный каталог сервера

Пример работы с сервером OnDemand:

```
//*****
import java.util.*;
import java.io.*;
import com.ibm.edms.od.*;

public class TcServerMisc
{
    public static void main ( String argv[] )
    {
        ODServer odServer;
        String str;
        int j;

        //-----
        // Если указано недостаточное число параметров, отображается синтаксис и работа завершается
        //-----
        if ( argv.length < 4 )
        {
            System.out.println( "формат: java TcServerMisc <сервер>
<ИД-пользователя> <пароль> <каталог конфигурации>
<локальный каталог сервера>" );
            return;
        }

        try
        {
            //-----
            // Подготовка ко входу
            //-----
            System.out.println( "В этом примере выполняются следующие действия:" );
```

```

System.out.println( " Применение методов setServer, setUserId и setPassword класса ODServer" );
System.out.println( " для подготовки ко входу в систему" );
System.out.println( " Определение имени приложения" );
System.out.println( " Отображение" );
System.out.println( " имени локального каталога" );
System.out.println( " сервера" );
System.out.println( " ИД пользователя" );
System.out.println( " Пароля" );
System.out.println( " Типа соединения" );
System.out.println( " Определение и отображение номера порта" );
System.out.println( " Определение типа соединения" );
System.out.println( " Выход из системы" );
System.out.println( "" );
System.out.println( "Проверка правильности информации." );
System.out.println( "" );
System.out.println( "-----" );
System.out.println( "" );

//-----
// Вход на указанный сервер
//-----
odServer = new ODServer( );
odServer.initialize( argv[3], "TcServerMisc.java" );
odServer.setServer( argv[0] );
odServer.setUserId( argv[1] );
odServer.setPassword( argv[2] );

System.out.println( "Вход на сервер " + argv[0] + "..." );
if ( argv.length == 4 )
    odServer.logon( );
else
{
    if ( argv.length == 5 )
    {
        odServer.setLocalDir( argv[4] );
        odServer.logon( odServer.getServerName( ),
            odServer.getUserId( ),
            odServer.getPassword( ),
            ODConstant.CONNECT_TYPE_LOCAL,
            0,
            odServer.getLocalDir( ) );
    }
}

//-----
// Проверка различных методов
//-----
System.out.println( "Изменение имени приложения на TcServerMisc.java..." );
odServer.setApplicationName( "TcServerMisc.java" );

System.out.println( "Локальный каталог: " + odServer.getLocalDir( ) );
System.out.println( "Имя сервера: " + odServer.getServerName( ) );
System.out.println( "ИД пользователя: " + odServer.getUserId( ) );
System.out.println( "Пароль: " + odServer.getPassword( ) );
System.out.println( "Тип соединения: " + getConnectTypeName(odServer.getConnectType( ) ) );

j = odServer.getPort( );
System.out.println( "Изменение номера порта на " + j + "..." );
odServer.setPort( j );
System.out.println( "Порт: " + j );

if ( argv.length == 4 )
{
    System.out.println( "Изменение типа соединения на ODConstant.CONNECT_TYPE_TCPIP..." );
    odServer.setConnectType( ODConstant.CONNECT_TYPE_TCPIP );
}
else
{
    System.out.println( "Изменение типа соединения на ODConstant.CONNECT_TYPE_LOCAL..." );
    odServer.setConnectType( ODConstant.CONNECT_TYPE_LOCAL );
}

//-----
// Очистка
//-----
System.out.println( "Выход из системы..." );
odServer.logoff( );
odServer.terminate( );
System.out.println( "" );
System.out.println( "-----" );
System.out.println( "" );
System.out.println( "Проверка завершена - при необходимости выполните анализ результатов" );
System.out.println( "" );
}

catch ( ODException e )
{
    System.out.println( "ODException: " + e );
    System.out.println( " id = " + e.getErrorId( ) );
    System.out.println( " msg = " + e.getErrorMsg( ) );
    e.printStackTrace( );
}

catch ( Exception e2 )

```

```

    {
        System.out.println( "exception: " + e2 );
        e2.printStackTrace( );
    }
}

static String getConnectTypeName( char type )
{
    String str;

    switch( type )
    {
        case ODConstant.CONNECT_TYPE_TCPIP:
            str = "TCP/IP";
            break;
        case ODConstant.CONNECT_TYPE_LOCAL:
            str = "LOCAL";
            break;
        default:
            str = "*** Неизвестный тип соединения";
            break;
    }

    return str;
}
}

```

Просмотр списка групп приложений папки

Объект класса `ODFolder` представляет папку `OnDemand`.

В следующем примере с помощью методов `ODFolder` отображается число групп приложений для поиска в папке, а также отображается список имен этих групп.

В данном примере применяются следующие методы `ODFolder`:

- `getNumApplGroups`
- `getApplGroups`
- `close`

Кроме того, в этом примере применяются методы `ODServer` для подготовки входа в систему, открытия указанной папки и выхода. В данном примере применяются следующие методы `ODServer`:

- `initialize`
- `logon`
- `openFolder`
- `logoff`
- `terminate`

В данном примере применяются следующие параметры выполнения:

- Имя сервера
- Ид пользователя
- Пароль
- Имя папки
- Каталог конфигурации (расположение файла `arswww.ini`)
- (дополнительно) Локальный каталог сервера

Пример, позволяющий просмотреть список групп приложений, расположенных в указанной папке:

```

//*****
import java.util.*;
import java.io.*;
import com.ibm.edms.od.*;

public class TcApplGrp
{
    public static void main ( String argv[] )
    {
        ODServer odServer;
        ODFolder odFolder;
        Object[] appl_grps;
        int j;

        //-----

```

```

// Если указано недостаточное число параметров, отображается синтаксис и работа завершается
//-----
if ( argv.length < 5 )
{
    System.out.println( "формат: java TcAppGrp <сервер>
<ИД пользователя> <пароль> <папка> <каталог конфигурации>
[<локальный каталог сервера>]" );
    return;
}

try
{
    //-----
    // Подготовка ко входу в систему
    //-----
    System.out.println( "В этом примере выполняются следующие действия:" );
    System.out.println( "  Вход на указанный сервер" );
    System.out.println( "  Открытие указанного каталога" );
    System.out.println( "  Отображение имени папки" );
    System.out.println( "  Отображение числа групп приложений" );
    System.out.println( "  Отображение списка имен групп приложений" );
    System.out.println( "" );
    System.out.println( "-----" );

    //-----
    // Вход на указанный сервер
    //-----
    odServer = new ODServer( );
    odServer.initialize( argv[4], "TcListCriteria.java" );

    System.out.println( "Вход на сервер " + argv[0] + "..." );
    if ( argv.length == 5 )
        odServer.logon( argv[0], argv[1], argv[2] );
    else
        if ( argv.length == 6 )
            odServer.logon( argv[0], argv[1], argv[2], ODConstant.CONNECT_TYPE_LOCAL, 0, argv[5] );

    //-----
    // Открытие указанного каталога
    //-----
    System.out.println( "Открывается каталог " + argv[3] + "..." );
    odFolder = odServer.openFolder( argv[3] );

    //-----
    // Отображение числа групп приложений и списка их имен
    //-----
    System.out.println( "В папке содержатся " + odFolder.getNumAppGroups( ) + " групп приложений:" );
    appl_grps = odFolder.getAppGroups( );
    for ( j = 0; j < appl_grps.length; j++ )
        System.out.println( " " + appl_grps[j].toString( ) );

    //-----
    // Очистка
    //-----
    odFolder.close( );
    odServer.logoff( );
    odServer.terminate( );
    System.out.println( "" );
    System.out.println( "-----" );
);

    System.out.println( "" );
    System.out.println( "Пример завершен - при необходимости выполните анализ результатов" );
    System.out.println( "" );
}

catch ( ODEException e )
{
    System.out.println( "ODEException: " + e );
    System.out.println( "  id = " + e.getErrorId( ) );
    System.out.println( "  msg = " + e.getErrorMsg( ) );
    e.printStackTrace( );
}

catch ( Exception e2 )
{
    System.out.println( "exception: " + e2 );
    e2.printStackTrace( );
}
}
}

```

Поиск папки

Объект класса `ODFolder` представляет папку `OnDemand`. Объект класса `ODCriteria` представляет критерии поиска для папки `OnDemand`. Объект класса `ODHit` представляет документ `OnDemand`.

В следующем примере с помощью методов `ODFolder` открывается указанная папка, отображается ее имя, описание, порядок отображения и критерии поиска, выполняется поиск в этой папке, после чего она закрывается. В данном примере

применяются методы `ODCriteria` для указания текущего операнда поиска и значений поиска. Кроме того, в этом примере применяются методы `ODHit` для получения отображаемых значений документа, типа документа, постоянного идентификатора документа, расположения документа, а также типа данных MIME документа.

В данном примере применяются следующие методы `ODFolder`:

- `getName`
- `getDescription`
- `getDisplayOrder`
- `getCriteria`
- `search`
- `getSearchMessage`
- `close`

В данном примере применяются следующие методы `ODCriteria`:

- `getName`
- `setOperand`
- `setSearchValue`
- `setSearchValues`

В данном примере применяются следующие методы `ODHit`:

- `getDisplayValue`
- `getDisplayValues`
- `getDocType`
- `getMimeType`
- `getDocLocation`
- `getDocId`

Кроме того, в этом примере применяются методы `ODServer` для подготовки входа в систему, открытия указанной папки и выхода. В данном примере применяются следующие методы `ODServer`:

- `initialize`
- `logon`
- `openFolder`
- `terminate`

В данном примере применяются следующие параметры выполнения:

- Имя сервера
- Ид пользователя
- Пароль
- Имя папки
- Имя критериев
- Оператор (один из следующих: `eq`, `ne`, `lt`, `le`, `gt`, `ge`, `in`, `ni`, `li`, `nl`, `be`, `nb`)
- Поисковое значение 1
- (дополнительно) Поисковое значение 2
- Каталог конфигурации (расположение файла `arswww.ini`)

Примечание: Число совпадений можно ограничить с помощью параметра `MAXHITS` в файле `arswww.ini`.

Пример поиска в папке:

```
//*****  
import java.util.*;  
import java.io.*;  
import com.ibm.edms.od.*;  
  
public class TcSearch  
{  
    public static void main ( String argv[] )  
    {
```

```

ODServer odServer;
ODFolder odFolder;
ODCriteria odCrit;
ODHit odHit;
Enumeration values_enum;
Vector hits;
String[] display_crit;
String header, line1, line2, hit_value, useable_value;
boolean mismatch_detected;
int j, k, opr;

//-----
// Если указано недостаточное число параметров, отображается синтаксис и работа завершается
//-----
if ( argv.length < 9 )
{
    System.out.println( "формат: java TcSearch <сервер>
<ИД пользователя> <пароль> <папка> <критерии>
<оператор> <значение1> <значение2>
<каталог конфигурации>" );
    return;
}

try
{
    //-----
    // Подготовка ко входу в систему
    //-----
    System.out.println( "В этом примере выполняются следующие действия: " );
    System.out.println( "  Вход на указанный сервер" );
    System.out.println( "  Открытие указанного каталога" );
    System.out.println( "  Отображение имени папки и ее описания" );
    System.out.println( "  Получение указанных критериев" );
    System.out.println( "  Определение оператора" );
    System.out.println( "  Определение операндов" );
    System.out.println( "  Поиск в папке" );
    System.out.println( "  Отображение сообщения поиска (если оно существует)" );
    System.out.println( "  Отображение числа совпадений" );
    System.out.println( "  Отображение списка найденных документов, в
котором для каждого совпадения выделено три строки:" );
    System.out.println( "    1. Значения совпадения, возвращенные методом ODHit.getDisplayValue" );
    System.out.println( "    2. Значения совпадения, возвращенные методом ODHit.getDisplayValues" );
    System.out.println( "    3. Тип документа, тип Mime, расположение документа, ИД документа" );
    System.out.println( " " );
    System.out.println( "Проверка совпадения строк 1 и 2 списка найденных документов, а также значений" );
    System.out.println( "списка найденных документов с результатами поиска с помощью клиента Windows." );
    System.out.println( "Если число совпадений ограничено в файле argswww.ini, то число совпадений может быть" );
    System.out.println( "меньше, чем полученное с помощью клиента Windows." );
    System.out.println( " " );
    System.out.println( "-----" );
    System.out.println( " " );

    //-----
    // Вход на указанный сервер
    //-----
    odServer = new ODServer( );
    odServer.initialize( argv[8], "TcSearch.java" );
    System.out.println( "Вход на сервер " + argv[0] + "..." );
    odServer.logon( argv[0], argv[1], argv[2] );

    //-----
    // Открытие указанного каталога и поиск запрашиваемых критериев
    //-----
    System.out.println( "Открывается каталог " + argv[3] + "..." );
    odFolder = odServer.openFolder( argv[3] );
    System.out.println( "Имя=" + odFolder.getName( ) + " Описание=" +
odFolder.getDescription( ) + " " );
    System.out.println( "Получение критериев " + argv[4] + "..." );
    odCrit = odFolder.getCriteria( argv[4] );

    //-----
    // Преобразование параметра оператора во внутреннее значение оператора и определение
    // оператора критериев
    //-----
    System.out.println( "Изменение оператора на " + argv[5] + "..." );
    if ( argv[5].equals( "eq" ) )
        opr = ODConstant.OPEqual;
    else if ( argv[5].equals( "ne" ) )
        opr = ODConstant.OPNotEqual;
    else if ( argv[5].equals( "lt" ) )
        opr = ODConstant.OPLessThan;
    else if ( argv[5].equals( "le" ) )
        opr = ODConstant.OPLessThanEqual;
    else if ( argv[5].equals( "gt" ) )
        opr = ODConstant.OPGreaterThan;
    else if ( argv[5].equals( "ge" ) )
        opr = ODConstant.OPGreaterThanEqual;
    else if ( argv[5].equals( "in" ) )
        opr = ODConstant.OPIn;
    else if ( argv[5].equals( "ni" ) )
        opr = ODConstant.OPNotIn;
    else if ( argv[5].equals( "li" ) )
        opr = ODConstant.OPLike;
    else if ( argv[5].equals( "nl" ) )
        opr = ODConstant.OPNotLike;
    else if ( argv[5].equals( "be" ) )
        opr = ODConstant.OPBetween;
    else if ( argv[5].equals( "nb" ) )
        opr = ODConstant.OPNotBetween;
    else
        opr = -1;

    System.out.println( "Определение операндов..." );
    odCrit.setOperand( opr );

    if ( opr == ODConstant.OPBetween || opr == ODConstant.OPNotBetween )
    {
        odCrit.setSearchValues( argv[6], argv[7] );
        System.out.println( " " + odCrit.getName( ) + " " + getOperatorName( opr ) + " " + argv[6] + " и " + argv[7] );
    }
}

```

```

else
{
    odCrit.setSearchValue( argv[6] );
    System.out.println( " " + odCrit.getName( ) + " " + getOperatorName( opr ) + " " + argv[6] );
}

//-----
// Поиск в нанке
//-----
System.out.println( " Поиск " + argv[3] + "..." );
hits = odFolder.search( );
System.out.println( " Сообщение поиска: " + odFolder.getSearchMessage( ) );
System.out.println( " Число совпадений: " + hits.size( ) );

//-----
// Отображение совпадений
//-----
mismatch_detected = false;
if ( hits != null && hits.size( ) > 0 )
{
    display_crit = odFolder.getDisplayOrder( );
    header = " ";
    for( j = 0; j < display_crit.length; j++ )
        header = header + display_crit[j] + "--";
    System.out.println( " -----" );
    System.out.println( header + " (from ODHit.getDisplayValue method)" );
    System.out.println( header + " (from ODHit.getDisplayValues method)" );
    System.out.println( " DocType--MimeType--DocLocation--DocId" );
    System.out.println( " -----" );
    for ( j = 0; j < hits.size( ); j++ )
    {
        odHit = (ODHit)hits.elementAt( j );
        line1 = " ";
        for ( k = 0; k < display_crit.length; k++ )
        {
            hit_value = odHit.getDisplayValue( display_crit[k] );
            useable_value = ( hit_value.equals( "" ) ) ? " " : hit_value;
            line1 = line1 + useable_value + "--";
        }
        System.out.println( line1 );
        line2 = " ";
        for ( values_enum = odHit.getDisplayValues( ); values_enum.hasMoreElements( ); )
        {
            hit_value = (String)values_enum.nextElement( );
            useable_value = ( hit_value.equals( "" ) ) ? " " : hit_value;
            line2 = line2 + useable_value + "--";
        }
        System.out.println( line2 );
        System.out.println( " " + getDocTypeString( odHit.getDocType( ) ) +
            "--" + odHit.getMimeType( ) +
            "--" + getLocationString( odHit.getDocLocation( ) ) +
            "--" + odHit.getDocId( ) );
        if ( !line1.equals( line2 ) )
            mismatch_detected = true;
    }
}

//-----
// Очистка
//-----
odFolder.close( );
odServer.logoff( );
odServer.terminate( );
System.out.println( "" );
System.out.println( "-----" );
System.out.println( "" );
System.out.println( "Проверка завершена - при необходимости выполните анализ
результатов" );
System.out.println( "" );
if ( mismatch_detected )
{
    System.out.println( "*** Найдено по крайней мере одно несоответствие между" );
    System.out.println( "*** первой и второй строками совпадения" );
    System.out.println( "" );
}
}

catch ( ODEException e )
{
    System.out.println( "ODEException: " + e );
    System.out.println( " id = " + e.getErrorId( ) );
    System.out.println( " msg = " + e.getErrorMsg( ) );
    e.printStackTrace( );
}

catch ( Exception e2 )
{
    System.out.println( "исключение: " + e2 );
    e2.printStackTrace( );
}
}

static String getOperatorName( int oper )
{
    String str;

    switch( oper )
    {
        case ODConstant.OPEqual:
            str = "Равно";
            break;
        case ODConstant.OPNotEqual:
            str = "Не равно";
            break;
        case ODConstant.OPLessThan:
            str = "Меньше";
            break;
        case ODConstant.OPLessThanEqual:
            str = "Меньше или равно";
            break;
    }
}

```

```

case ODCConstant.OPGreaterThan:
    str = "Больше";
    break;
case ODCConstant.OPGreaterThanEqual:
    str = "Больше или равно";
    break;
case ODCConstant.OPIn:
    str = "Входит в состав";
    break;
case ODCConstant.OPNotIn:
    str = "Не входит в состав";
    break;
case ODCConstant.OPLike:
    str = "Похоже на";
    break;
case ODCConstant.OPNotLike:
    str = "Не похоже на";
    break;
case ODCConstant.OPBetween:
    str = "Лежит в диапазоне";
    break;
case ODCConstant.OPNotBetween:
    str = "Не лежит в диапазоне";
    break;
default:
    str = "Неизвестный оператор";
    break;
}

return str;
}

static String getDocTypeString( char type )
{
    String str;

    switch( type )
    {
        case ODCConstant.FileTypeAFP:
            str = "AFP";
            break;
        case ODCConstant.FileTypeBMP:
            str = "BMP";
            break;
        case ODCConstant.FileTypeEMAIL:
            str = "EMAIL";
            break;
        case ODCConstant.FileTypeGIF:
            str = "GIF";
            break;
        case ODCConstant.FileTypeJFIF:
            str = "JFIF";
            break;
        case ODCConstant.FileTypeLINE:
            str = "LINE";
            break;
        case ODCConstant.FileTypeMETA:
            str = "META";
            break;
        case ODCConstant.FileTypeNONE:
            str = "NONE";
            break;
        case ODCConstant.FileTypePCX:
            str = "PCX";
            break;
        case ODCConstant.FileTypePDF:
            str = "PDF";
            break;
        case ODCConstant.FileTypePNG:
            str = "PNG";
            break;
        case ODCConstant.FileTypeTIFF:
            str = "TIFF";
            break;
        case ODCConstant.FileTypeUSRDEF:
            str = "USRDEF";
            break;
        default:
            str = "*** Недопустимый тип документа ***";
            break;
    }

    return str;
}

static String getLocationString( int loc )
{
    String str;

    switch( loc )
    {
        case ODCConstant.DocLocationCache:
            str = "Кэш";
            break;
        case ODCConstant.DocLocationArchive:
            str = "Архив";
            break;
        case ODCConstant.DocLocationExternal:
            str = "Внешний";
            break;
        case ODCConstant.DocLocationUnknown:
            str = "Неизвестно";
            break;
        default:
            str = "*** Недопустимое расположение документа ***";
            break;
    }

}

```

```
        return str;
    }
}
```

Поиск в папке с помощью строки SQL

В следующем примере с помощью методов `ODFolder` открывается указанная папка, выполняется поиск в этой папке с помощью указанной строки SQL, после чего она закрывается. В данном примере применяются методы `ODHit` для отображения числа элементов, соответствующих запросу, а также отображения списка найденных документов.

В данном примере применяются следующие методы `ODFolder`:

- `setAppGroupForSearchWithSQL`
- `search`
- `getDisplayOrder`
- `close`

В данном примере применяются следующие методы `ODHit`:

- `getDisplayValue`

Кроме того, в этом примере применяются методы `ODServer` для подготовки входа в систему, открытия указанной папки и отключения. В данном примере применяются следующие методы `ODServer`:

- `initialize`
- `logon`
- `openFolder`
- `logoff`
- `terminate`

В данном примере применяются следующие параметры выполнения:

- Имя сервера
- Ид пользователя
- Пароль
- Имя папки
- Имя группы приложений
- Строка SQL
- Каталог конфигурации (расположение файла `arswww.ini`)

Пример поиска в папке с помощью строки SQL:

```
//*****
//
// Пример: TcSearchWithSQL
//
// В данном примере выполняются следующие действия:
// Вход на указанный сервер
// Открытие указанного каталога
// Поиск в папке с помощью строки SQL
// Отображение числа совпадений
// Отображение списка найденных документов
//
// Выполняется проверка следующих методов:
// ODServer
//   initialize
//   logon
//   openFolder
//   logoff
//   terminate
// ODFolder
//   setAppGroupForSearchWithSQL
//   search
//   getDisplayOrder
//   close
// ODHit
//   getDisplayValue
//
// Параметры:
// 1. Имя сервера
// 2. ИД пользователя
// 3. Пароль
// 4. Имя папка
// 5. Имя группы приложений
```

```

// 6. Строка SQL
// 7. Каталог конфигурации (содержит arswww.ini)
//
//*****
import java.util.*;
import java.io.*;
import com.ibm.edms.od.*;

public class TcSearchWithSQL
{
    public static void main ( String argv[] )
    {
        ODServer odServer;
        ODFolder odFolder;
        ODHit odHit;
        Enumeration values_enum;
        Vector hits;
        String[] display_crit;
        String server, userid, password, folder, directory;
        String sql, appl_group;
        String header, line, hit_value, useable_value;
        int j, k;

        //-----
        // Если указано недостаточное число параметров, отображается синтаксис и работа завершается
        //-----
        if ( argv.length < 6 )
        {
            System.out.println( "формат: java TcSearchWithSQL <сервер>
<ИД пользователя> <пароль> <папка> <группа приложений>
<строка sql> <каталог настройки>" );
            return;
        }

        try
        {
            //-----
            // Set the stage
            //-----
            System.out.println( "В этом примере выполняются следующие действия:" );
            System.out.println( " Вход на указанный сервер" );
            System.out.println( " Открытие указанного каталога" );
            System.out.println( " Поиск в каталоге с помощью указанной строки SQL" );
            System.out.println( " Отображение числа совпадений" );
            System.out.println( " Отображение списка найденных документов" );
            System.out.println( "" );
            System.out.println( "-----" );
            System.out.println( "" );

            //-----
            // Вход на указанный сервер
            //-----
            server = argv[0];
            userid = argv[1];
            password = argv[2];
            folder = argv[3];
            appl_group = argv[4];
            sql = argv[5];
            directory = argv[6];

            odServer = new ODServer( );
            odServer.initialize( directory, "TcSearchWithSQL.java" );
            System.out.println( "Вход на сервер " + server + "..." );
            odServer.logon( server, userid, password );

            //-----
            // Открытие указанной папки
            //-----
            System.out.println( "Открытие папки " + folder + " ..." );
            odFolder = odServer.openFolder( folder );

            //-----
            // Поиск в папке
            //-----
            if ( appl_group.length( ) > 0 )
            {
                System.out.println( "Указание группы приложений для поиска: " + appl_group );
                odFolder.setAppGroupForSearchWithSQL( appl_group );
            }

            //-----
            // Поиск в папке
            //-----
            System.out.println( "Поиск в папке " + folder + " ..." );
            hits = odFolder.search( sql );
            System.out.println( " Число совпадений: " + hits.size( ) );

            //-----
            // Отображение найденных документов
            //-----
            if ( hits != null && hits.size( ) > 0 )
            {
                display_crit = odFolder.getDisplayOrder( );
                header = " ";
                for( j = 0; j < display_crit.length; j++ )
                    header = header + display_crit[j] + "--";
                System.out.println( "
-----" );
                System.out.println( header );
                System.out.println( "
-----" );
                for ( j = 0; j < hits.size( ); j++ )
                {
                    odHit = (ODHit)hits.elementAt( j );
                    line = " ";
                    for ( k = 0; k < display_crit.length; k++ )
                    {
                        hit_value = odHit.getDisplayValue( display_crit[k] );
                        useable_value = ( hit_value.equals( "" ) ) ? " " : hit_value;
                        line = line + useable_value + "--";
                    }
                }
            }
        }
    }
}

```

```

        }
        System.out.println( line );
    }
}

//-----
// Очистка
//-----
odFolder.close( );
odServer.logoff( );
odServer.terminate( );
System.out.println( "" );
System.out.println( "-----" );
System.out.println( "" );
System.out.println( "Пример завершен - при необходимости выполните анализ результатов" );
System.out.println( "" );
}

catch ( ODEException e )
{
    System.out.println( "ODEException: " + e );
    System.out.println( "    id = " + e.getErrorId( ) );
    System.out.println( "    msg = " + e.getErrorMsg( ) );
    e.printStackTrace( );
}

catch ( Exception e2 )
{
    System.out.println( "exception: " + e2 );
    e2.printStackTrace( );
}
}
}

```

Отмена поиска

В следующем примере с помощью методов `ODServer.cancel` отменяется выполнение поиска.

В этом примере применяются методы `ODServer`, `ODFolder` и `ODCriteria` для входа на сервер, открытия папки и определения критерия даты 1970-2001. Затем запускается вторая нить, с помощью которой выполняется поиск. После завершения второй нити отображается число совпадений. Вторая нить запускается снова для выполнения поиска. Процесс приостанавливается на 5 секунд, после чего поиск отменяется. После завершения второй нити отображается число совпадений.

В данном примере применяются следующие методы `ODServer`:

- initialize
- logon
- openFolder
- logoff
- terminate

В данном примере применяются следующие методы `ODFolder`:

- getCriteria
- search
- close

В данном примере применяются следующие методы `ODCriteria`:

- setOperand
- setSearchValues

В данном примере применяются следующие параметры выполнения:

- Имя сервера
- Ид пользователя
- Пароль
- Имя папки
- Каталог конфигурации (расположение файла `arswww.ini`)
- (дополнительно) Локальный каталог сервера

Пример отмены поиска:

```

//*****
import java.util.*;
import java.io.*;

```

```

import com.ibm.edms.od.*;

class TestThread extends Thread
{
    ODFolder odFolder;

    TestThread( ODFolder fld )
    {
        odFolder = fld;
    }

    public void run( )
    {
        Vector hits;

        try
        {
            System.out.println( " Поиск с помощью второй нити..." );
            hits = odFolder.search( );
            System.out.println( " Поиск завершен - Число совпадений: " + hits.size( ) );
        }

        catch ( ODException e )
        {
            System.out.println( "ODException: " + e );
            System.out.println( " id = " + e.getErrorId( ) );
            System.out.println( " msg = " + e.getErrorMsg( ) );
            e.printStackTrace( );
        }

        catch ( Exception e2 )
        {
            System.out.println( "exception: " + e2 );
            e2.printStackTrace( );
        }
    }
}

public class TcCancelSearch
{
    public static void main ( String argv[] )
    {
        ODServer odServer;
        ODFolder odFolder;
        ODCriteria odCrit;
        TestThread search_thread;
        int j;

        //-----
        // Если указано недостаточное число параметров, отображается синтаксис и работа завершается
        //-----
        if ( argv.length < 5 )
        {
            System.out.println( "формат: java TcCancelSearch <сервер>
<ИД пользователя> <пароль> <папка>
<каталог конфигурации> [<локальный каталог сервера>]" );
            return;
        }

        try
        {
            //-----
            // Подготовка ко входу в систему
            //-----
            System.out.println( "В этом примере выполняются следующие действия:" );
            System.out.println( " Вход на указанный сервер" );
            System.out.println( " Открытие указанного каталога" );
            System.out.println( " Определение критерия даты 1970-2001" );
            System.out.println( " Запуск второй нити для выполнения поиска" );
            System.out.println( " После завершения второй нити отображается число совпадений" );
            System.out.println( " Запуск второй нити для выполнения поиска" );
            System.out.println( " Ожидание в течение 5 секунд" );
            System.out.println( " Отмена поиска" );
            System.out.println( " После завершения второй нити отображается число совпадений" );
            System.out.println( "" );
            System.out.println( "Проверка выбора папки, включающей в себя критерий Дата." );
            System.out.println( "Проверка того, что в папке содержится большое число совпадений и что файл argswww.ini" );
            System.out.println( "не ограничивает число возвращаемых совпадений." );
            System.out.println( "" );
            System.out.println( "-----" );
            System.out.println( "" );

            //-----
            // Вход на указанный сервер
            //-----
            odServer = new ODServer( );
            odServer.initialize( argv[4], "TcCancelSearch.java" );

            System.out.println( "Вход на сервер " + argv[0] + "..." );
            if ( argv.length == 5 )
                odServer.logon( argv[0], argv[1], argv[2] );
            else
                if ( argv.length == 6 )
                    odServer.logon( argv[0], argv[1], argv[2],
                    ODConstant.CONNECT_TYPE_LOCAL, 0, argv[5] );

            //-----
            // Открытие указанной папки и отображение ее имени и описания
            //-----
            System.out.println( "Открытие " + argv[3] + "..." );
            odFolder = odServer.openFolder( argv[3] );
            odCrit = odFolder.getCriteria( "Дата" );
            odCrit.setOperand( ODConstant.OPBetween );
        }
    }
}

```

```

odCrit.setSearchValues( "01/01/70", "01/01/01" );

//-----
// Поиск с помощью другой нити, кратковременное ожидание и отмена поиска
//-----
System.out.println( "Основная нить запускает поиск (без попытки отмены)..." );
search_thread = new TestThread( odFolder );
search_thread.start( );
search_thread.join( );

System.out.println( "Основная нить запускает поиск (будет выполнена отмена)..." );
search_thread = new TestThread( odFolder );
search_thread.start( );
System.out.println( "Основная нить ожидает в течение 5 секунд..." );
( Thread.currentThread( ) ).sleep( 500 );
System.out.println( "Основная нить пытается отменить поиск..." );
odServer.cancel( );
System.out.println( "Основная нить возвращена после попытки отмены" );
search_thread.join( );

//-----
// Очистка
//-----
odFolder.close( );
odServer.logoff( );
odServer.terminate( );
System.out.println( "" );
System.out.println( "-----" );
System.out.println( "" );
System.out.println( "Пример завершен - Убедитесь, что второй поиск," );
System.out.println( " который был отменен, возвратил меньшее число результатов, чем первый" );
System.out.println( "" );
}

catch ( ODEException e )
{
    System.out.println( "ODEException: " + e );
    System.out.println( " id = " + e.getErrorId( ) );
    System.out.println( " msg = " + e.getErrorMsg( ) );
    e.printStackTrace( );
}

catch ( Exception e2 )
{
    System.out.println( "exception: " + e2 );
    e2.printStackTrace( );
}
}
}

```

Просмотр списка критериев поиска

В следующем примере показано, каким образом с помощью методов `ODCriteria` можно просмотреть список критериев поиска указанной папки. В этом примере отображается список имен полей поиска, вместе с операторами по умолчанию, допустимыми операторами, типом полей, а также значениями поиска по умолчанию. Значения по умолчанию отображаются с помощью методов `ODCriteria.getSearchValues` и `ODCriteria.getValues`. Фиксированные значения поиска отображаются для полей поиска, определенных в качестве `FixedChoice` или `Segment`.

В данном примере применяются следующие методы `ODCriteria`:

- `setOperand`
- `getValidOperands`
- `getType`
- `getValues`
- `setSearchValues`
- `getFixedValues`

В данном примере применяются следующие методы `ODServer`:

- `initialize`
- `logon`
- `openFolder`
- `logoff`
- `terminate`

В данном примере применяются следующие методы `ODFolder`:

- `getCriteria`
- `close`

В данном примере применяются следующие параметры выполнения:

- Имя сервера
- Ид пользователя
- Пароль
- Имя папки
- Каталог конфигурации (расположение файла arswww.ini)
- (дополнительно) Локальный каталог сервера

Пример просмотра критериев поиска:

```

//*****
import java.util.*;
import java.io.*;
import com.ibm.edms.od.*;

public class TcListCriteria
{
    public static void main ( String argv[] )
    {
        ODServer odServer;
        ODFolder odFolder;
        ODCriteria odCrit;
        Enumeration crit_enum;
        Vector value_vec;
        String[] search_values, fixed_values;
        int[] valid_oprs;
        int j, opr;
        char field_type;

        //-----
        // Если указано недостаточное число параметров, отображается синтаксис и работа завершается
        //-----
        if ( argv.length < 5 )
        {
            System.out.println( "формат: java TcListCriteria <сервер>
            <Ид пользователя> <пароль> <папка>
            <каталог конфигурации>
            [<локальный каталог сервера>]" );
            return;
        }

        try
        {
            //-----
            // Подготовка ко входу в систему
            //-----
            System.out.println( "В этом примере выполняются следующие действия:" );
            System.out.println( "  Вход на указанный сервер" );
            System.out.println( "  Открытие указанного каталога" );
            System.out.println( "  Отображение имени папки и ее описания" );
            System.out.println( "  Отображение числа критериев папки" );
            System.out.println( "  Для каждого критерия отображается" );
            System.out.println( "    Имя" );
            System.out.println( "    Оператор по умолчанию" );
            System.out.println( "    Допустимые операторы" );
            System.out.println( "    Тип поля" );
            System.out.println( "    Значения по умолчанию (с помощью метода ODCrit.getSearchValues)" );
            System.out.println( "    Значения по умолчанию (с помощью метода ODCrit.getValues)" );
            System.out.println( "    Фиксированные значения (только для критериев FixedChoice и Segment)" );
            System.out.println( "" );
            System.out.println( "Проверка того, что операторы не обозначены в качестве 'Неизвестных операторов'," );
            System.out.println( "типы полей не обозначаются в качестве 'Неизвестных типов'," );
            System.out.println( "значения по умолчанию совпадают для всех методов, а также" );
            System.out.println( "с помощью клиента Windows отображается аналогичная информация." );
            System.out.println( "" );
            System.out.println( "-----" );
            System.out.println( "" );

            //-----
            // Вход на указанный сервер
            //-----
            odServer = new ODServer( );
            odServer.initialize( argv[4], "TcListCriteria.java" );

            System.out.println( "Вход на сервер " + argv[0] + "..." );
            if ( argv.length == 5 )
                odServer.logon( argv[0], argv[1], argv[2] );
            else
                if ( argv.length == 6 )
                    odServer.logon( argv[0], argv[1], argv[2], ODConstant.CONNECT_TYPE_LOCAL, 0, argv[5] );

            //-----
            // Открытие указанной папки и отображение ее имени и описания
            //-----
            System.out.println( "Открывается каталог " + argv[3] + "..." );
            odFolder = odServer.openFolder( argv[3] );
            System.out.println( "Имя=" + odFolder.getName( ) + " Описание=" + odFolder.getDescription( ) + " " );
            System.out.println( "Применяются следующие " +
            odFolder.getNumCriteria( ) + " критериев:" );

            //-----
            // Для каждого критерия папки,
            //-----
            for ( crit_enum = odFolder.getCriteria( ); crit_enum.hasMoreElements( ); )
            {
                //-----
            }
        }
    }
}

```

```

// Отображается имя критерия
//-----
System.out.println( "" );
odCrit = (ODCriteria)crit_enum.nextElement( );
System.out.println( odCrit.getName( ) );

//-----
// Отображается оператор по умолчанию
//-----
opr = odCrit.getOperand( );
System.out.println( " Оператор по умолчанию: " );
System.out.println( " " + getOperatorName( opr ) );

//-----
// Отображаются допустимые операторы
//-----
valid_oprs = odCrit.getValidOperands( );
System.out.println( " Допустимые операторы: " );
for ( j = 0; j < valid_oprs.length; j++ )
    System.out.println( " " + getOperatorName( valid_oprs[j] ) );

//-----
// Отображается тип поля
//-----
field_type = odCrit.getType( );
System.out.println( " Тип: " );
System.out.println( " " + getTypeName( field_type ) );

//-----
// Отображаются значения по умолчанию с помощью метода ODCrit.getValues( )
//-----
value_vec = odCrit.getValues( );
System.out.println( " Значения по умолчанию (метод ODCrit.getValues):" );
System.out.println( " " + value_vec.elementAt( 0 ) + " " );
System.out.println( " " + value_vec.elementAt( 1 ) + " " );

//-----
// Отображаются значения по умолчанию с помощью метода ODCrit.getSearchValues( )
//-----
search_values = odCrit.getSearchValues( );
System.out.println( " Значения по умолчанию (метод ODCrit.getSearchValues):" );
for ( j = 0; j < search_values.length; j++ )
    System.out.println( " " + search_values[j] + " " );

//-----
// Отображаются фиксированные значения
//-----
switch ( field_type )
{
    case ODConstant.InputTypeChoice:
    case ODConstant.InputTypeSegment:
        fixed_values = odCrit.getFixedValues( );
        System.out.println( " Фиксированные значения (только для полей типа FixedChoice и Segment):" );
        for ( j = 0; j < fixed_values.length; j++ )
            System.out.println( " " + fixed_values[j] + " " );
        break;
}

//-----
// Очистка
//-----
odFolder.close( );
odServer.logoff( );
odServer.terminate( );
System.out.println( " " );
System.out.println( "-----" );
System.out.println( " " );
System.out.println( "Пример завершен - при необходимости выполните анализ результатов" );
System.out.println( " " + e.getErrorMsg( ) );
System.out.println( " " );
}

catch ( ODEException e )
{
    System.out.println( "ODEException: " + e );
    System.out.println( " id = " + e.getErrorId( ) );
    System.out.println( " msg = " + e.getErrorMsg( ) );
    e.printStackTrace( );
}

catch ( Exception e2 )
{
    System.out.println( "исключительная ситуация: " + e2 );
    e2.printStackTrace( );
}

}

static String getOperatorName( int oper )
{
    String str;

    switch( oper )
    {
        case ODConstant.OPEqual:
            str = "Равно";
            break;
        case ODConstant.OPNotEqual:
            str = "Не равно";
            break;
        case ODConstant.OPLessThan:
            str = "Меньше";
            break;
    }
}

```

```

        case ODCConstant.OPLessThanEqual:
            str = "Меньше или равно";
            break;
        case ODCConstant.OPGreaterThan:
            str = "Больше";
            break;
        case ODCConstant.OPGreaterThanEqual:
            str = "Больше или равно";
            break;
        case ODCConstant.OPIn:
            str = "Входит в состав";
            break;
        case ODCConstant.OPNotIn:
            str = "Не входит в состав";
            break;
        case ODCConstant.OPLike:
            str = "Похоже на";
            break;
        case ODCConstant.OPNotLike:
            str = "Не похоже на";
            break;
        case ODCConstant.OPBetween:
            str = "Лежит в диапазоне";
            break;
        case ODCConstant.OPNotBetween:
            str = "Не лежит в диапазоне";
            break;
        default:
            str = "*** Неизвестный оператор";
            break;
    }

    return str;
}

static String getTypeName( char type )
{
    String str;

    switch( type )
    {
        case ODCConstant.InputTypeNormal:
            str = "Normal";
            break;
        case ODCConstant.InputTypeTextSearch:
            str = "TextSearch";
            break;
        case ODCConstant.InputTypeNoteTextSearch:
            str = "NoteTextSearch";
            break;
        case ODCConstant.InputTypeNoteColor:
            str = "NoteColor";
            break;
        case ODCConstant.InputTypeChoice:
            str = "FixedChoice";
            break;
        case ODCConstant.InputTypeSegment:
            str = "Segment";
            break;
        default:
            str = "*** Неизвестный тип";
            break;
    }

    return str;
}
}

```

Просмотр списка папок с информацией о них

В следующем примере с помощью методов `ODServer` распечатывается число папок указанного сервера, в которых можно выполнить поиск от имени указанного ИД пользователя. Для каждой папки в отдельной строке распечатывается ее имя и описание.

В данном примере применяются следующие методы `ODServer`:

- initialize
- logon
- getNumFolders
- getFolderNames
- getFolderDescription
- logoff
- terminate

В данном примере применяются следующие параметры выполнения:

- Имя сервера
- Ид пользователя

- Пароль
- Каталог конфигурации (расположение файла `arswww.ini`)
- (дополнительно) Локальный каталог сервера

Пример, позволяющий просмотреть список папок и информацию о них:

```

//*****
import java.util.*;
import java.io.*;
import com.ibm.edms.od.*;

public class TcListFolders
{
    public static void main ( String argv[] )
    {
        ODServer    odServer;
        Enumeration folders_enum;
        String folder_name, folder_desc;
        int num_folders;

        //-----
        // Если указано недостаточное число параметров, отображается синтаксис и работа завершается
        //-----
        if ( argv.length < 4 )
        {
            System.out.println( "формат: java TcListFolders <сервер>
<ИД-пользователя> <пароль> <каталог конфигурации>
<локальный каталог сервера>" );
            return;
        }

        try
        {
            //-----
            // Подготовка ко входу в систему
            //-----
            System.out.println( "В этом примере выполняются следующие действия:" );
            System.out.println( "  Отображается число папок сервера, доступных заданному ИД пользователя" );
            System.out.println( "  Для каждой папки в отдельной строке отображается ее имя и описание" );
            System.out.println( "  " );
            System.out.println( "Данная информация должна совпадать с результатами работы клиента Windows" );
            System.out.println( "(если выбран переключатель 'Все'), но последовательность папок" );
            System.out.println( "может отличаться в зависимости от указанного сервера" );
            System.out.println( "  " );
            System.out.println( "-----" );
            System.out.println( "  " );

            //-----
            // Вход на указанный сервер
            //-----
            odServer = new ODServer( );
            odServer.initialize( argv[3], "TcListFolders.java" );

            System.out.println( "Вход на сервер " + argv[0] + "..." );
            if ( argv.length == 4 )
                odServer.logon( argv[0], argv[1], argv[2] );
            else
                if ( argv.length == 5 )
                    odServer.logon( argv[0], argv[1], argv[2], ODConstant.CONNECT_TYPE_LOCAL, 0, argv[4] );

            //-----
            // Отображение числа доступных папок.
            //-----
            num_folders = odServer.getNumFolders( );
            System.out.println( "  " );
            System.out.println( "На сервере " + argv[0] + " для пользователя " + argv[1] +
" доступно " + num_folders + " папок:" );

            //-----
            // Отображение имен папок и их описаний
            //-----
            for ( folders_enum = odServer.getFolderNames( ); folders_enum.hasMoreElements( ); )
            {
                folder_name = (String)folders_enum.nextElement( );
                folder_desc = odServer.getFolderDescription( folder_name );
                System.out.println( "  " + folder_name + " --- " + folder_desc );
            }

            //-----
            // Очистка
            //-----
            odServer.logoff( );
            odServer.terminate( );
            System.out.println( "  " );
            System.out.println( "-----" );
            System.out.println( "  " );
            System.out.println( "Пример завершен - вы можете сравнить результаты с клиентом Windows" );
            System.out.println( "  " );
        }

        catch ( ODException e )
        {
    }
}

```

```

        System.out.println( "ODException: " + e );
        System.out.println( "   id = " + e.getErrorId( ) );
        System.out.println( "   msg = " + e.getErrorMsg( ) );
        e.printStackTrace( );
    }

    catch ( Exception e2 )
    {
        System.out.println( "исключительная ситуация: " + e2 );
        e2.printStackTrace( );
    }
}
}

```

Просмотр списка документов

В следующем примере с помощью методов `ODFolder` и `ODHit` выполняется поиск в папке с критериями по молчанию, отображается число найденных документов и отображается список этих документов.

В данном примере применяются следующие методы `ODFolder`:

- `getName`
- `getDisplayOrder`
- `search`
- `close`

В данном примере применяются следующие методы `ODHit`:

- `getDisplayValue`

В данном примере применяются следующие методы `ODServer`:

- `initialize`
- `logon`
- `openFolder`
- `logoff`
- `terminate`

В данном примере применяются следующие параметры выполнения:

- Имя сервера
- Ид пользователя
- Пароль
- Имя папки
- Каталог конфигурации (расположение файла `arswww.ini`)

Пример просмотра списка документов:

```

//*****
import java.util.*;
import java.io.*;
import com.ibm.edms.od.*;

public class TcSortedHitlist
{
    public static void main ( String argv[] )
    {
        ODServer odServer;
        ODFolder odFolder;
        ODHit odHit;
        Vector hits;
        String[] display_crit;
        String server, userid, password, folder, value;
        int j, k;

        //-----
        // Если указано недостаточное число параметров, отображается синтаксис и работа завершается
        //-----
        if ( argv.length < 5 )
        {
            System.out.println( "формат: java TcSortedHitlist <сервер>
            <ИД пользователя> <пароль> <папка> <каталог конфигурации>" );
            return;
        }
    }
}

```

```

try
{
    //-----
    // Подготовка ко входу в систему
    //-----
    System.out.println( "В этом примере выполняются следующие действия:" );
    System.out.println( " Вход на указанный сервер" );
    System.out.println( " Открытие указанной папки" );
    System.out.println( " Поиск в папке с помощью критерия по умолчанию" );
    System.out.println( " Отображение сообщения поиска (при необходимости)" );
);
    System.out.println( " Отображение числа совпадений" );
    System.out.println( " Отображение списка найденных документов" );
    System.out.println( "" );
    System.out.println( "-----" );
    System.out.println( "" );

    //-----
    // Вход на указанный сервер
    //-----
    server = argv[0];
    userid = argv[1];
    password = argv[2];
    folder = argv[3];
    odServer = new ODServer( );
    odServer.initialize( argv[4], "TcSortedHitlist.java" );
    System.out.println( "Вход на сервер " + server + " от имени " + userid + "/" + password + "..." );
    odServer.logon( server, userid, password );

    //-----
    // Открытие папки и поиск в ней
    //-----
    System.out.println( "Открытие " + folder + "..." );
    odFolder = odServer.openFolder( folder );
    System.out.println( "Поиск в папке с помощью критериев по умолчанию..." );
    hits = odFolder.search( );
    System.out.println( " Число совпадений: " + hits.size( ) );

    //-----
    // Отображение совпадений
    //-----
    if ( hits != null && hits.size( ) > 0 )
    {
        display_crit = odFolder.getDisplayOrder( );
        value = " ";
        for( j = 0; j < display_crit.length; j++ )
            value = value + display_crit[j] + " ";
        System.out.println( value );
        for ( j = 0; j < hits.size( ); j++ )
        {
            odHit = (ODHit)hits.elementAt( j );
            value = " ";
            for ( k = 0; k < display_crit.length; k++ )
                value = value + odHit.getDisplayValue( display_crit[k] ) + " ";
            System.out.println( value );
        }
    }

    //-----
    // Очистка
    //-----
    odFolder.close( );
    odServer.logoff( );
    odServer.terminate( );
    System.out.println( "" );
    System.out.println( "-----" );
    System.out.println( "" );
    System.out.println( "Пример завершен - Убедитесь, что порядок папок" );
    System.out.println( " совпадает с результатами работы клиента Windows" );
    System.out.println( "" );
}

catch ( ODEException e )
{
    System.out.println( "ODEException: " + e );
    System.out.println( " id = " + e.getErrorId( ) );
    System.out.println( " msg = " + e.getErrorMsg( ) );
    e.printStackTrace( );
}

catch ( Exception e2 )
{
    System.out.println( "исключительная ситуация: " + e2 );
}

```

```

        e2.printStackTrace( );
    }
}
}

```

Загрузка документа

В следующем примере показаны три различных способа загрузки документов:

- ODServer
- ODFolder
- ODHit

В данном примере выполняются следующие действия: вход на сервер, открытие указанной папки, поиск в папке с помощью критериев по умолчанию, отображение числа совпадений, загрузка данных первого найденного документа с помощью `ODHit.retrieve`, `ODServer.retrieve` и `ODFolder.retrieve`. В этом примере отображается длина данных, загруженных с помощью различных методов, и показываются результаты сравнения этих значений.

В данном примере применяются следующие методы `ODServer`:

- initialize
- logon
- openFolder
- retrieve
- logoff
- terminate

В данном примере применяются следующие методы `ODFolder`:

- search
- retrieve
- close

В данном примере применяются следующие методы `ODHit`:

- getDocId
- retrieve

В данном примере применяются следующие параметры выполнения:

- Имя сервера
- Ид пользователя
- Пароль
- Имя папки
- Каталог конфигурации (расположение файла `arswww.ini`)
- (дополнительно) Локальный каталог сервера

Пример загрузки документа:

```

//*****
import java.util.*;
import java.io.*;
import com.ibm.edms.od.*;

public class TcRetrieve
{
    public static void main ( String argv[] )
    {
        ODServer odServer;
        ODFolder odFolder;
        ODHit odHit;
        TcCallback callback;
        Vector hits;
        Vector hit_to_retrieve;
        byte[] data_from_hit;
        byte[] data_from_server;
        byte[] data_from_folder;
        int j;

        //-----
        // Если указано недостаточное число параметров, отображается синтаксис и работа завершается
        //-----
    }
}

```

```

if ( argv.length < 5 )
{
    System.out.println( "формат: java TcRetrieve <сервер>
<ИД пользователя> <пароль> <папка>
<каталог конфигурации> [<локальный каталог сервера>]" );
    return;
}

try
{
    //-----
    // Подготовка ко входу в систему
    //-----
    System.out.println( "В этом примере выполняются следующие действия:" );
    System.out.println( " Вход на указанный сервер" );
    System.out.println( " Открытие указанной папки" );
    System.out.println( " Поиск в папке с помощью критерия по умолчанию" );
    System.out.println( " Отображение числа совпадений" );
    System.out.println( " Загрузка данных первого найденного документа с помощью ODHit.retrieve" );
    System.out.println( " Загрузка данных первого найденного документа с помощью ODServer.retrieve" );
    System.out.println( " Загрузка данных первого найденного документа с помощью ODFolder.retrieve" );
    System.out.println( " Отображение длины данных, полученных с помощью каждого метода" );
    System.out.println( " Сравнение длины и данных, полученных с помощью каждого метода" );
    System.out.println( " Отображение результатов сравнения" );
    System.out.println( "" );
    System.out.println( "-----" );
    System.out.println( "" );

    //-----
    // Вход на указанный сервер
    //-----
    odServer = new ODServer( );
    odServer.initialize( argv[4], "TcRetrieve.java" );
    System.out.println( "Вход на сервер " + argv[0] + "..." );
    if ( argv.length == 5 )
        odServer.logon( argv[0], argv[1], argv[2] );
    else
        odServer.logon( argv[0], argv[1], argv[2], ODConstant.CONNECT_TYPE_LOCAL, 0, argv[5] );

    //-----
    // Открытие указанной папки и поиск с помощью критериев по умолчанию
    //-----
    System.out.println( "Открывается каталог " + argv[3] + " ..." );
    odFolder = odServer.openFolder( argv[3] );
    System.out.println( "Поиск с помощью критериев по умолчанию..." );
    hits = odFolder.search( );
    System.out.println( "Число совпадений: " + hits.size( ) );
    //-----
    // Загрузка данных и их сравнение
    //-----
    if ( hits.size( ) > 0 )
    {
        odHit = (ODHit)hits.elementAt( 0 );
        System.out.println( "Загрузка данных первого найденного документа с помощью
ODHit.retrieve..." );
        data_from_hit = odHit.retrieve( "" );
        System.out.println( "Загрузка данных первого найденного документа с помощью
ODServer.retrieve..." );
        data_from_server = odServer.retrieve( odHit.getDocId( ), argv[3], "" );
        hit_to_retrieve = new Vector( );
        hit_to_retrieve.addElement( odHit );
        System.out.println( "Загрузка данных первого найденного документа с помощью
ODFolder.retrieve (применяется обратный вызов)..." );
        callback = new TcCallback( );
        odFolder.retrieve( hit_to_retrieve, callback );
        data_from_folder = callback.getData( );
        System.out.println( "Length of data from:" );
        System.out.println( " ODHit.retrieve=" + data_from_hit.length );
        System.out.println( " ODServer.retrieve=" + data_from_server.length );
        System.out.println( " ODFolder.retrieve=" + data_from_folder.length );
        if ( data_from_hit.length == data_from_server.length )
        {
            for ( j = 0; j < data_from_hit.length; j++ )
            {
                if ( data_from_hit[j] != data_from_server[j] )
                    break;
            }
            if ( j == data_from_hit.length )
            {
                System.out.println( "Сравнение ODHit и ODServer: Длина и содержимое совпадают" );
                if ( data_from_hit.length == data_from_folder.length )
                {
                    for ( j = 0; j < data_from_folder.length; j++ )
                    {
                        if ( data_from_hit[j] != data_from_folder[j] )
                            break;
                    }
                    if ( j == data_from_folder.length )
                        System.out.println( "Сравнение ODHit и ODFolder: Длина и содержимое совпадают" );
                    else
                    {
                        System.out.println( "*** Сравнение ODHit и ODFolder:
Несоответствие данных для смещения " + j );
                        System.out.println( " Данные ODHit: " + data_from_hit[j] );
                        System.out.println( " Данные ODFolder: " + data_from_folder[j] );
                    }
                }
            }
            else
                System.out.println( "*** Сравнение ODHit и ODFolder: Не совпадает длина" );
        }
    }
}

```

```

        else
        {
            System.out.println( "*** Сравнение ODHit и ODServer:
            Несоответствие данных для смещения " + j );
            System.out.println( " Данные ODHit: " + data_from_hit[j] );
            System.out.println( " Данные ODServer: " + data_from_server[j] );
        }
    }
    else
        System.out.println( "*** Сравнение ODHit и ODServer: Не совпадает длина" );
}
else
    System.out.println( "Документ недоступен для загрузки" );

//-----
// Очистка
//-----
odFolder.close( );
odServer.logoff( );
odServer.terminate( );
System.out.println( "" );
System.out.println( "-----" );
System.out.println( "" );
System.out.println( "Пример завершен - выполните анализ результатов сравнения" );
System.out.println( "" );
System.out.println( "Если для типа данных в файле arswm.ini указано значение 'native', то" );
System.out.println( "все длины и данные должны совпадать.
В противном случае могут возникать отличия. " );
System.out.println( "" );
}

catch ( ODException e )
{
    System.out.println( "ODException: " + e );
    System.out.println( " id = " + e.getErrorId( ) );
    System.out.println( " msg = " + e.getErrorMsg( ) );
    e.printStackTrace( );
}

catch ( Exception e2 )
{
    System.out.println( "исключительная ситуация: " + e2 );
    e2.printStackTrace( );
}
}
}

```

В следующем примере для загрузки документа используются методы `ODCallback`.

```

//*****
import java.util.*;
import java.io.*;
import com.ibm.edms.od.*;

public class TcCallback extends ODCallback
{
    byte[] data_from_folder;
    boolean init = true;

    TcCallback( )
    {
    }

    public void HitHandleCallback( int hit, int off, int len )
    {
    }

    public boolean HitCallback( String docid, char type, String[] values )
        throws Exception
    {
        return true;
    }

    public boolean DataCallback( byte[] data )
    {
        byte[] temp;
        int j, k;

        //-----
        // Если загружается первый блок данных, то инициализуется контейнер.
        // В противном случае новые данные добавляются к предыдущим.
        //-----
        if ( init )
        {
            data_from_folder = data;
            init = false;
        }
        else
        {
        }
    }
}

```

```

        temp = new byte[ data_from_folder.length + data.length ];
        for ( j = 0; j < data_from_folder.length; j++ )
            temp[j] = data_from_folder[j];
        k = data_from_folder.length;
        for ( j = 0; j < data.length; j++ )
            temp[k++] = data[j];
        data_from_folder = temp;
    }

    return true;
}

public byte[] getData( )
{
    return data_from_folder;
}
}

```

Печать документа

В следующем примере применяются методы `ODServer` и `ODFolder` для просмотра списка принтеров, доступных для сервера, а также печати документа на указанном принтере. Кроме того, в этом примере применяются методы `ODServer` для подготовки входа в систему, открытия указанной папки и отключения.

В данном примере применяются следующие методы `ODServer`:

- `initialize`
- `logon`
- `openFolder`
- `getServerPrinters`
- `logoff`
- `terminate`

В данном примере применяются следующие методы `ODFolder`:

- `search`
- `printDocs`
- `close`

В данном примере применяются следующие параметры выполнения:

- Имя сервера
- ИД пользователя
- Пароль
- Имя папки
- Имя принтера
- Каталог конфигурации (расположение файла `arswww.ini`)
- (дополнительно) Локальный каталог сервера

Пример печати документа:

```

//*****
import java.util.*;
import java.io.*;
import com.ibm.edms.od.*;

public class TcPrintHit
{
    public static void main ( String argv[] )
    {
        ODServer odServer;
        ODFolder odFolder;
        ODHit odHit;
        Vector hits, hit_to_print;
        String [] printers;
        String printer_name;
        boolean match;
        int j;

        //-----
        // Если указано недостаточное число параметров, отображается синтаксис и работа завершается
        //-----
        if ( argv.length < 6 )
        {
            System.out.println( "формат: java TcPrintHit <сервер>
<ИД пользователя> <пароль> <папка> <принтер>

```

```

<каталог конфигурации> [<каталог локального сервера>]" );
return;
}

try
{
//-----
// Подготовка ко входу в систему
//-----
System.out.println( "В этом примере выполняются следующие действия:" );
System.out.println( " Вход на указанный сервер" );
System.out.println( " Отображение списка принтеров, доступных для сервера" );
System.out.println( " Открытие указанной папки" );
System.out.println( " Поиск в папке с помощью критерия по умолчанию" );
System.out.println( " Отображение числа совпадений" );
System.out.println( " Печать первого найденного документа на указанном принтере" );
System.out.println( "" );
System.out.println( "-----" );
System.out.println( "" );

//-----
// Вход на указанный сервер
//-----
odServer = new ODServer( );
odServer.initialize( argv[5], "TcPrintHit.java" );
System.out.println( "Logging on to " + argv[0] + "..." );
if ( argv.length == 6 )
    odServer.logon( argv[0], argv[1], argv[2] );
else
    odServer.logon( argv[0], argv[1], argv[2], ODConstant.CONNECT_TYPE_LOCAL, 0, argv[6] );

//-----
// Если для сервера доступны принтеры
//-----
System.out.println( "Загрузка списка принтеров сервера..." );
printer_name = argv[4];
printers = odServer.getServerPrinters( );
if ( printers.length > 0 )
{
//-----
// Отображение списка доступных принтеров сервера
//-----
System.out.println( "Доступны " + printers.length + " принтеров:" );
match = false;
for( j = 0; j < printers.length; j++ )
{
    System.out.println( " " + printers[j] );
    if ( printers[j].equals( printer_name ) )
        match = true;
}

if ( match )
{
//-----
// Открытие указанной папки и поиск с помощью критериев по умолчанию
//-----
System.out.println( "Открывается каталог " + argv[3] + "..." );
odFolder = odServer.openFolder( argv[3] );
System.out.println( "Поиск с помощью критериев по умолчанию..." );
hits = odFolder.search( );
System.out.println( " Число совпадений: " + hits.size( ) );

//-----
// Печать первого найденного документа на указанном принтере сервера
//-----
if ( hits.size( ) > 0 )
{
    hit_to_print = new Vector( );
    odHit = (ODHit)hits.elementAt( 0 );
    hit_to_print.addElement( odHit );
    System.out.println( "Печать первого совпадения на принтере " + printer_name + "..." );
    odFolder.printDocs( hit_to_print, printer_name );
}
else
    System.out.println( "Документ недоступен для печати" );

odFolder.close( );
}
else
    System.out.println( "Указанный принтер ( " + printer_name + " ) недоступен" );
}
else
    System.out.println( "Для этого сервера отсутствуют доступные принтеры" );

//-----
// Очистка
//-----
odServer.logoff( );
odServer.terminate( );
System.out.println( "" );
System.out.println( "-----" );
System.out.println( "" );
System.out.println( "Проверка завершена - выполните анализ результатов" );
System.out.println( "" );
}

catch ( ODEException e )
{
System.out.println( "ODEException: " + e );
System.out.println( " id = " + e.getErrorId( ) );
System.out.println( " msg = " + e.getErrorMsg( ) );
e.printStackTrace( );
}

catch ( Exception e2 )
{
System.out.println( "исключительная ситуация: " + e2 );
}

```

```
        e2.printStackTrace( );
    }
}
```

Просмотр информации о заметках

В следующем примере с помощью методов `ODNote` отображается подробная информация о заметке. В данном примере выполняются следующие действия: вход на сервер, открытие указанной папки, поиск в папке с помощью критериев по умолчанию, отображение числа совпадений, отображение числа заметок, связанных с первым документом, а также подробной информации о всех заметках, связанных с документом. Эта информация включает в себя расположение заметки на странице документа, цвет фона, дату и время добавления заметки, ИД пользователя, создавшего ее и другие атрибуты.

В данном примере применяются следующие методы `ODNote`:

- `getColor`
- `getDateTime`
- `getGroupName`
- `getOffsetX`
- `getOffsetY`
- `getPageNum`
- `getText`
- `getUserid`
- `isOkToCopy`
- `isPublic`

В данном примере применяются следующие методы `ODServer`:

- `initialize`
- `logon`
- `openFolder`
- `logoff`
- `terminate`

В данном примере применяются следующие методы `ODFolder`:

- `search`
- `close`

В данном примере применяются следующие методы `ODHit`:

- `getNotes`

В данном примере применяются следующие параметры выполнения:

- Имя сервера
- Ид пользователя
- Пароль
- Имя папки
- Каталог конфигурации (расположение файла `arswww.ini`)
- (дополнительно) Локальный каталог сервера

Пример просмотра информации о заметках:

```
//*****
import java.util.*;
import java.io.*;
import com.ibm.edms.od.*;

public class TcListNotes
{
    public static void main ( String argv[] )
    {
        ODServer odServer;
        ODFolder odFolder;
        ODHit odHit;
        ODNote odNote;
        Vector hits, notes;
```

```

int j;

//-----
// Если указано недостаточное число параметров, отображается синтаксис и работа завершается
//-----
if ( argv.length < 5 )
{
    System.out.println( "формат: java TcListNotes <сервер>
<ИД пользователя> <пароль> <папка>
<каталог конфигурации> [<локальный каталог сервера>" );
    return;
}

try
{
    //-----
    // Подготовка ко входу в систему
    //-----
    System.out.println( "В этом примере выполняются следующие действия:" );
    System.out.println( "  Вход на указанный сервер" );
    System.out.println( "  Открытие указанной папки" );
    System.out.println( "  Поиск в папке с помощью критерия по умолчанию" );
    System.out.println( "  Отображение числа совпадений" );
    System.out.println( "  Отображение числа заметок, связанных с первым найденным документом" );
    System.out.println( "  Отображение информации о каждой заметке" );
    System.out.println( "  " );
    System.out.println( "-----" );
    System.out.println( "  " );

    //-----
    // Вход на указанный сервер
    //-----
    odServer = new ODServer( );
    odServer.initialize( argv[4], "TcListNotes.java" );
    System.out.println( "Вход на сервер " + argv[0] + "..." );
    if ( argv.length == 5 )
        odServer.logon( argv[0], argv[1], argv[2] );
    else
        odServer.logon( argv[0], argv[1], argv[2],
ODConstant.CONNECT_TYPE_LOCAL, 0, argv[6] );

    //-----
    // Открытие указанной папки и поиск с помощью критериев по умолчанию
    //-----
    System.out.println( "Открывается каталог " + argv[3] + "..." );
    odFolder = odServer.openFolder( argv[3] );
    System.out.println( "Поиск с помощью критериев по умолчанию..." );
    hits = odFolder.search( );
    System.out.println( "  Число совпадений: " + hits.size( ) );

    //-----
    // Отображение информации о всех заметках, добавленных в первый найденный документ
    //-----
    if ( hits.size( ) > 0 )
    {
        odHit = (ODHit)hits.elementAt( 0 );
        notes = odHit.getNotes( );
        System.out.println( "  Для первого найденного документа определено "
+ notes.size( ) + " заметок" );
        for ( j = 0; j < notes.size( ); j++ )
        {
            odNote = (ODNote)notes.elementAt( j );
            System.out.println( "    " + (j+1) + ". Текст=" + odNote.getText( ) + " " );
            System.out.println( "    ИД пользователя=" + odNote.getUserId( ) );
            System.out.println( "    Страница=" + odNote.getPageNum( ) );
            System.out.println( "    Цвет=" + odNote.getColor( ) );
            System.out.println( "    Дата=" + odNote.getDate( ) );
            System.out.println( "    Группа=" + odNote.getGroupName( ) );
            System.out.println( "    Смещение=" + odNote.getOffsetX( ) + " " );
            System.out.println( "    " + odNote.getOffsetY( ) + " " );
            System.out.println( "    OkToCopy=" + odNote.isOkToCopy( ) );
            System.out.println( "    Public=" + odNote.isPublic( ) );
        }
    }
    else
        System.out.println( "Документ недоступен - невозможно отобразить заметки" );

    //-----
    // Очистка
    //-----
    odFolder.close( );
    odServer.logoff( );
    odServer.terminate( );
    System.out.println( "  " );
    System.out.println( "-----" );
    System.out.println( "  " );
    System.out.println( "Пример завершен - Убедитесь, что информация" );
    System.out.println( "  совпадает с результатами работы клиента Windows" );
    System.out.println( "  " );
}

catch ( ODException e )
{
    System.out.println( "ODException: " + e );
    System.out.println( "  id = " + e.getErrorId( ) );
    System.out.println( "  msg = " + e.getErrorMsg( ) );
    e.printStackTrace( );
}

catch ( Exception e2 )
{

```

```

        System.out.println( "исключительная ситуация: " + e2 );
        e2.printStackTrace( );
    }
}
}

```

Добавление заметки

Объект класса `ODHit` представляет документ `OnDemand`. В следующем примере с помощью методов `ODHit` отображается число заметок, связанных с документом и добавляется новая заметка со следующими атрибутами:

- Указанный текст заметки
- `OkToCopy=false`
- `Public=false` (личная заметка)
- Имя группы не указано

В данном примере применяются следующие методы `ODHit`:

- `getNotes`
- `addNote`

Кроме того, в этом примере применяются методы `ODServer` для подготовки входа в систему, открытия указанной папки и выхода, а также методы `ODFolder` для поиска в папке, получения числа совпадений и закрытия папки. В данном примере применяются следующие методы `ODServer`:

- `initialize`
- `logon`
- `openFolder`
- `logoff`
- `terminate`

В данном примере применяются следующие методы `ODFolder`:

- `search`
- `getHits`
- `close`

В данном примере применяются следующие параметры выполнения:

- Имя сервера
- Ид пользователя
- Пароль
- Имя папки
- Текст заметки
- Каталог конфигурации (расположение файла `arswww.ini`)
- (дополнительно) Локальный каталог сервера

Пример добавления заметки:

```

//*****
import java.util.*;
import java.io.*;
import com.ibm.edms.od.*;

public class TcAddNote
{
    public static void main ( String argv[] )
    {
        ODServer odServer;
        ODFolder odFolder;
        ODHit odHit;
        ODNote odNote;
        Vector hits, notes;
        int j;

        //-----
        // Если указано недостаточное число параметров, отображается синтаксис и работа завершается
        //-----
        if ( argv.length < 6 )
            System.out.println( "формат: java TcAddNote <сервер>
            <Ид пользователя> <пароль> <папка> <текст заметки>
            <каталог конфигурации> [<локальный каталог сервера>" );
        {
            return;
        }
    }
}

```

```

try
{
//-----
// Подготовка ко входу в систему
//-----
System.out.println( "В этом примере выполняются следующие действия:" );
System.out.println( " Вход на указанный сервер" );
System.out.println( " Открытие указанной папки" );
System.out.println( " Поиск в папке с помощью критерия по умолчанию" );
System.out.println( " Отображение числа совпадений" );
System.out.println( " Отображение числа заметок, связанных с первым найденным документом" );
System.out.println( " Добавление новой заметки со следующими атрибутами:" );
System.out.println( "   Указанный текст заметки" );
System.out.println( "   OkToCopy=false" );
System.out.println( "   Public=false (личная заметка)" );
System.out.println( "   Пустое имя группы" );
System.out.println( "   " );
System.out.println( "-----" );
System.out.println( "   " );

//-----
// Вход на указанный сервер
//-----
odServer = new ODServer( );
odServer.initialize( argv[5], "TcAddNote.java" );
System.out.println( "Вход на сервер " + argv[0] + "..." );
if ( argv.length == 6 )
    odServer.logon( argv[0], argv[1], argv[2] );
else
    odServer.logon( argv[0], argv[1], argv[2], ODConstant.CONNECT_TYPE_LOCAL, 0, argv[6] );

//-----
// Открытие указанной папки и поиск с помощью критериев по умолчанию
//-----
System.out.println( "Открывается каталог " + argv[3] + "..." );
odFolder = odServer.openFolder( argv[3] );
System.out.println( "Поиск с помощью критериев по умолчанию..." );
odFolder.search( );
hits = odFolder.getHits( );
System.out.println( " Число совпадений: " + hits.size( ) );

//-----
// Добавление новой заметки
//-----
if ( hits.size( ) > 0 )
{
    odHit = (ODHit)hits.elementAt( 0 );
    notes = odHit.getNotes( );
    System.out.println( " Для первого найденного документа определено "
        + notes.size( ) + " заметок" );

    odNote = new ODNote( );
    odNote.setText( argv[4] );
    odNote.setGroupName( "" );
    odNote.setOkToCopy( false );
    odNote.setPublic( false );

    System.out.println( " Добавление заметки со следующими параметрами:" );
    System.out.println( "   Text=" + odNote.getText( ) + " " );
    System.out.println( "   OkToCopy=" + odNote.isOkToCopy( ) );
    System.out.println( "   Public=" + odNote.isPublic( ) );
    System.out.println( "   Group=" + odNote.getGroupName( ) );

    odHit.addNote( odNote );
}
else
    System.out.println( "Документ недоступен - невозможно отобразить заметки" );

//-----
// Очистка
//-----
odFolder.close( );
odServer.logoff( );
odServer.terminate( );
System.out.println( " " );
System.out.println( "-----" );
System.out.println( " " );
System.out.println( "Пример завершен - Убедитесь, что новая заметка добавлена правильным образом." );
System.out.println( " Для этого просмотрите ее с помощью клиента Windows" );
System.out.println( " " );
}

catch ( ODException e )
{
    System.out.println( "ODException: " + e );
    System.out.println( "   id = " + e.getErrorId( ) );
    System.out.println( "   msg = " + e.getErrorMsg( ) );
    e.printStackTrace( );
}

catch ( Exception e2 )
{
    System.out.println( "исключительная ситуация: " + e2 );
    e2.printStackTrace( );
}
}
}

```

Обновление документа

В следующем примере показано, каким образом можно обновить документ.

В этом примере с помощью методов `ODServer`, `ODFolder` и `ODCriteria` выполняются следующие действия: подключение к серверу с указанными ИД пользователя и паролем, открытие указанной папки, указание значений для двух полей поиска, указание в поле поиска Дата нулевого значения и поиск в папке. Для документа, который будет возвращен в результате запроса, обновляются значения базы данных с помощью методов `ODHit`.

В данном примере применяются следующие методы `ODServer`:

- `initialize`
- `logon`
- `openFolder`
- `logoff`
- `terminate`

В данном примере применяются следующие методы `ODFolder`:

- `getName`
- `getDisplayOrder`
- `getCriteria`
- `search`
- `closeinitialize`

В данном примере применяются следующие методы `ODCriteria`:

- `setOperand`
- `setSearchValue`

В данном примере применяются следующие методы `ODHit`:

- `getDisplayValue`
- `update`

Кроме того, применяются следующие параметры выполнения:

- Имя сервера
- Ид пользователя
- Пароль
- Имя папки
- Имя критерия 1
- Поисковое значение 1
- Имя критерия 2
- Поисковое значение 2
- Новое поисковое значение, заменяющее поисковое значение 2
- Каталог конфигурации (расположение файла `arswww.ini`)

Пример обновления документа:

```
//*****
import java.util.*;
import java.io.*;
import com.ibm.edms.od.*;

public class TcUpdate
{
    public static void main ( String argv[] )
    {
        ODServer odServer;
        ODFolder odFolder;
        ODCriteria odCrit;
        ODHit odHit;
        Hashtable hash;
        Vector hits;
        String[] display_crit;
        String line, crit1, crit2, value1, value2, new_value;
        int j;

        //-----
        // Если указано недостаточное число параметров, отображается синтаксис и работа завершается
        //-----
    }
}
```

```

if ( argv.length < 10 )
{
    System.out.println( "формат: java TcUpdate <сервер>
<ИД пользователя> <пароль> <папка>
<критерий1> <значение1> +
"<критерий2> <значение2> <новое значение2>" );
    return;
}

try
{
    System.out.println( "В этом примере выполняются следующие действия:" );
    System.out.println( " Вход на указанный сервер" );
    System.out.println( " Открытие указанной папки" );
    System.out.println( " Задание поисковых значений" );
    System.out.println( " Поиск в папке" );
    System.out.println( " Изменение значения второго критерия для первого найденного документа" );
    System.out.println( " на новое значение" );
    System.out.println( "" );
    System.out.println( "Проверка внесенных изменений с помощью клиента Windows." );
    System.out.println( "" );
    System.out.println( "-----" );
    System.out.println( "" );

    //-----
    // Вход на указанный сервер
    //-----
    odServer = new ODServer( );
    odServer.initialize( argv[9], "TcUpdate.java" );
    System.out.println( "Logging on to " + argv[0] + "..." );
    odServer.logon( argv[0], argv[1], argv[2] );

    //-----
    // Открытие указанной папки и определение запрашиваемых критериев
    //-----
    crit1 = argv[4];
    crit2 = argv[6];
    value1 = argv[5];
    value2 = argv[7];
    new_value = argv[8];
    System.out.println( "Открывается каталог " + argv[3] + " ..." );
    odFolder = odServer.openFolder( argv[3] );
    odCrit = odFolder.getCriteria( crit1 );
    odCrit.setOperand( ODConstant.OPEqual );
    odCrit.setSearchValue( value1 );
    odCrit = odFolder.getCriteria( crit2 );
    odCrit.setOperand( ODConstant.OPEqual );
    odCrit.setSearchValue( value2 );

    //-----
    // Поиск в папке
    //-----
    System.out.println( " Поиск " + crit1 + " = " + value1 + " и " + crit2 + " = " + value2 + "..." );
    hits = odFolder.search( );

    //-----
    // Если обнаружено по крайней мере одно совпадение
    //-----
    if ( hits != null && hits.size( ) > 0 )
    {
        //-----
        // Отображение значений для первого найденного документа
        //-----
        System.out.println( " Для первого найденного документа:" );
        line = " ";
        display_crit = odFolder.getDisplayOrder( );
        for( j = 0; j < display_crit.length; j++ )
            line = line + display_crit[j] + " ";
        System.out.println( line );
        line = " ";
        odHit = (ODHit)hits.elementAt( 0 );
        for ( j = 0; j < display_crit.length; j++ )
            line = line + odHit.getDisplayValue( display_crit[j] ) + " ";
        System.out.println( line );

        //-----
        // Создание хэш-таблицы всех существующих пар критерий/значение, кроме критерия 2,
        // для которого будет указано новое значение. Обновление значений совпадения
        //-----
        System.out.println( " Замена " + crit2 + " = " + value2 + " на " + crit2 + " = " + new_value );
        hash = new Hashtable( );
        for ( j = 0; j < display_crit.length; j++ )
        {
            if ( display_crit[j].equals( crit2 ) )
                hash.put( display_crit[j], new_value );
            else
                hash.put( display_crit[j], odHit.getDisplayValue( display_crit[j] ) );
        }
        odHit.update( hash );
    }
}
else
    System.out.println( "Совпадения не были обнаружены" );

```

```

//-----
// Очистка
//-----
odFolder.close( );
odServer.logoff( );
odServer.terminate( );
System.out.println( "" );
System.out.println( "-----" );
System.out.println( "" );
System.out.println( "Пример завершен - Запуск клиента Windows для" );
System.out.println( " проверки внесенных изменений." );
System.out.println( "" );
}

catch ( ODEException e )
{
System.out.println( "ODEException: " + e );
System.out.println( " id = " + e.getErrorId( ) );
System.out.println( " msg = " + e.getErrorMsg( ) );
e.printStackTrace( );
}

catch ( Exception e2 )
{
System.out.println( "exception: " + e2 );
e2.printStackTrace( );
}
}
}

```

Изменение пароля

В следующем примере с помощью метода `changePassword` класса `ODServer` изменяется пароль указанного пользователя. Кроме того, в этом примере применяются методы `ODServer` для подготовки входа в систему и выхода из нее.

В данном примере применяются следующие методы `ODServer`:

- `initialize`
- `logon`
- `changePassword`
- `logoff`
- `terminate`

В данном примере применяются следующие параметры выполнения:

- Имя сервера
- Ид пользователя
- Пароль
- Новый пароль
- Каталог конфигурации (расположение файла `arswww.ini`)
- (дополнительно) Локальный каталог сервера

Пример изменения пароля:

```

//*****
import java.util.*;
import java.io.*;
import com.ibm.edms.od.*;

public class TcChangePassword
{
public static void main ( String argv[] )
{
ODServer odServer;
String server, userid, original_password, new_password;

//-----
// Если указано недостаточное число параметров, отображается синтаксис и работа завершается
//-----
if ( argv.length < 5 )
{
System.out.println( "формат: java TcChangePassword <сервер>
<Ид пользователя> <пароль> <новый пароль>
<каталог конфигурации> [<локальный каталог сервера>]" );
return;
}

try
{
//-----
// Подготовка ко входу в систему
//-----

```

```

System.out.println( "В этом примере выполняются следующие действия:" );
System.out.println( "  Вход на сервер с помощью указанного пароля" );
System.out.println( "  Изменение пароля на новый" );
System.out.println( "  Выход из системы" );
System.out.println( "  Вход на сервер с помощью нового пароля" );
System.out.println( "  Восстановление старого пароля" );
System.out.println( "  Выход из системы" );
System.out.println( "" );
System.out.println( "Если пример выполняется без исключительных ситуаций, то дополнительный анализ");
System.out.println( "не требуется." );
System.out.println( "" );
System.out.println( "-----" );
System.out.println( "" );

//-----
// Создание указанного сервера
//-----
server = argv[0];
userid = argv[1];
original_password = argv[2];
new_password = argv[3];
odServer = new ODServer( );
odServer.initialize( argv[4], "TcChangePassword.java" );

//-----
// Вход на указанный сервер с помощью исходного пароля
//-----
System.out.println( "Вход на сервер " + server + " с помощью исходного пароля..." );
if ( argv.length == 5 )
    odServer.logon( server, userid, original_password );
else
    if ( argv.length == 6 )
        odServer.logon( server, userid, original_password, ODConstant.CONNECT_TYPE_LOCAL, 0, argv[5] );

//-----
// Изменение пароля и выход из системы
//-----
System.out.println( "Изменение пароля..." );
odServer.changePassword( new_password );
System.out.println( "Выход из системы..." );
odServer.logoff( );

//-----
// Вход на сервер с помощью нового пароля
//-----
System.out.println( "Вход на сервер " + server + " с помощью нового пароля..." );
if ( argv.length == 5 )
    odServer.logon( server, userid, new_password );
else
    if ( argv.length == 6 )
        odServer.logon( server, userid, new_password, ODConstant.CONNECT_TYPE_LOCAL, 0, argv[5] );

//-----
// Восстановление исходного пароля и выход из системы
//-----
System.out.println( "Восстановление исходного пароля..." );
odServer.changePassword( original_password );
System.out.println( "Выход из системы..." );
odServer.logoff( );

//-----
// Очистка
//-----
odServer.terminate( );
System.out.println( "" );
System.out.println( "-----" );
System.out.println( "" );
System.out.println( "Пример выполнен успешно" );
System.out.println( "" );
}

catch ( ODEException e )
{
    System.out.println( "ODEException: " + e );
    System.out.println( "  id = " + e.getErrorId( ) );
    System.out.println( "  msg = " + e.getErrorMsg( ) );
    e.printStackTrace( );
}

catch ( Exception e2 )
{
    System.out.println( "исключительная ситуация: " + e2 );
    e2.printStackTrace( );
}
}
}

```

Приложение Е. Преобразование AFP в HTML

Процесс преобразования AFP в HTML позволяет преобразовывать документы и ресурсы AFP в формат HTML. Для работы процесса преобразования AFP в HTML необходима служебная программа AFP2WEB Transform, предоставленная IBM Printing Systems Division. Администратор устанавливает и настраивает программу AFP2WEB Transform на сервере HTTP. Более подробную информацию о служебной программе AFP2WEB Transform можно получить в представительстве IBM. Кроме того, для документов AFP и ресурсов, обрабатываемых с помощью AFP2WEB Transform, необходимо указать опции конфигурации. В этом разделе приведены инструкции по указанию опций конфигурации.

Примечание: В данном документе AFP2HTML.INI является именем файла конфигурации. Для того чтобы указать файл, содержащий опции конфигурации, обратитесь к разделу “CONFIGFILE” на стр. 21.

Файл AFP2HTML.INI содержит опции конфигурации для AFP2WEB Transform. Как правило в файле AFP2HTML.INI указываются опции конфигурации для конкретных приложений AFP. Кроме того, вы можете указать набор опций по умолчанию. Для приложений AFP, не указанных в файле AFP2HTML.INI, в процессе преобразования документов и ресурсов программа AFP2WEB Transform применяет опции по умолчанию. Более подробная информация о процессе преобразования и соответствующих опциях приведена в документации по AFP2WEB Transform.

В следующих разделах приведена дополнительная информация о файле AFP2HTML.INI:

- Формат файла AFP2HTML.INI
- Опции AFP2WEB Transform
- Просмотр преобразованных документов

Примечание: Для преобразования документов с помощью апплета AFP2HTML следует также указать параметр AFPVIEWING=HTML в разделе DEFAULT BROWSER (или в разделах других браузеров) файла ARSWWW.INI. Подробная информация приведена в разделе “AFPVIEWING” на стр. 31. (Если вы планируете работать с API Загрузить документ, укажите параметр _аfr=HTML. Дополнительная информация приведена в разделе “Загрузить документ” на стр. 77.) Кроме того, необходимо указать каталог, в котором содержатся программы AFP2WEB Transform (см. “CONFIGFILE” на стр. 21).

Формат файла AFP2HTML.INI

Ниже приведен пример файла AFP2HTML.INI:

```
[CREDIT-CREDIT]
UseApplet=FALSE
ScaleFactor=1.0
CreateGIF=TRUE
SuppressFonts=FALSE
FontMapFile=creditFontMap.cfg
ImageMapFile=creditImageMap.cfg
```

```
[default]
ScaleFactor=1.0
```

```
CreateGIF=TRUE
SuppressFonts=FALSE
FontMapFile=fontmap.cfg
ImageMapFile=imagemap.cfg
```

Структура данного файла аналогична структуре файлов INI операционной системы Windows и содержит по одному разделу для каждого приложения AFP, а также один раздел по умолчанию. Заголовок раздела определяет группу приложений и приложение. Например, строка заголовка

```
[CREDIT-CREDIT]
```

Определяет группу приложений CREDIT и приложение CREDIT. Имена в строке заголовка разделяются с помощью символа – (дефис). Имена должны совпадать с именами групп приложений и приложений, определенными на сервере OnDemand. Если группа приложений содержит несколько приложений, то для каждого приложения следует создать отдельный раздел.

Опции раздела [default] предназначены для обработки документов приложений AFP, не указанных в файле AFP2HTML.INI. Кроме того, значения по умолчанию применяются, если в разделе приложения AFP не указана одна из опций.

Опция UseApplet является директивой для ODWEK. Она позволяет для просмотра вывода AFP2WEB Transform применять апплет AFP2HTML. Значение по умолчанию - TRUE. Если указано значение FALSE (апплет AFP2HTML не применяется для просмотра вывода), то вывод форматируется Web-браузером и отображается в его окне.

Остальные пять опций являются директивами для AFP2WEB Transform. В разделе “Опции AFP2WEB Transform” приведено краткое описание их применения в AFP2WEB Transform.

Опции AFP2WEB Transform

В разделе Табл. 14 перечислены опции, предназначенные для преобразования документов с помощью AFP2WEB Transform, которые можно указать в файле AFP2HTML.INI.

Таблица 14. Опции AFP2WEB Transform

Опция файла AFP2HTML.INI	Описание
AllObjects	Определяет способ обработки документов, сохраняемых в виде больших объектов на сервере OnDemand. Значение по умолчанию - 0 (ноль). Оно означает, что ODWEK загружает только первый сегмент документа. Если указано значение 1, то перед отправкой документа клиенту ODWEK загружает и преобразует все сегменты документа. Примечание: Применение поддержки больших документов может привести к значительным задержкам при просмотре очень больших документов на клиенте.
ScaleFactor	Позволяет изменить масштаб вывода в соответствии с указанным коэффициентом. Значение по умолчанию - 1.0. Например, если указать ScaleFactor=2.0, то вывод будет в два раза больше размера по умолчанию. Значение ScaleFactor=0.5 позволяет уменьшить вывод в два раза. Размер по умолчанию указывается с помощью параметра Масштаб на странице Логические представления приложения OnDemand.

Таблица 14. Опции AFP2WEB Transform (продолжение)

Опция файла AFP2HTML.INI	Описание
SuppressFonts	Определяет необходимость преобразования текстовых строк AFP. Если указано значение SuppressFonts=TRUE, то текст, для которого применяется шрифт, заданный в файле преобразования шрифтов, не преобразуется. Значение по умолчанию - FALSE. Оно означает, что преобразуются все текстовые строки AFP. Файл преобразования шрифтов задается с помощью опции FontMapFile.
FontMapFile	Задаёт полный путь к файлу преобразования шрифтов. В файле преобразования шрифтов содержится список шрифтов, требующих специальной обработки. Файл преобразования шрифтов по умолчанию называется imagfont.cfg и расположен в каталоге, в котором содержатся программы AFP2WEB Transform. Подробная информация о файле преобразования шрифтов приведена в документации по AFP2WEB Transform.
ImageMapFile	Задаёт файл преобразования изображений. Файл преобразования изображений позволяет удалять изображения из вывода, улучшать качество темных изображений, заменять существующие изображения на изображения, созданные AFP2WEB Transform. Преобразование изображений, часто встречающихся в документах AFP (например, логотипа компании) позволяет ускорить процесс преобразования документов. Если файл преобразования изображений указан, то он должен существовать и располагаться в каталоге, в котором содержатся программы AFP2WEB Transform. Подробная информация о файле преобразования изображений приведена в документации по AFP2WEB Transform.

Примечание: В процессе преобразования документов ODWEK отправляет программе AFP2WEB Transform следующие опции. Эти опции не указываются в файле AFP2HTML.INI.

- Ориентация. Определяет значение поворота, применяемое при просмотре документа. По умолчанию применяется значение, указанное в параметре Ориентация на странице Просмотр информации приложения OnDemand.
- Цвет изображения. Определяет цвет, применяемый для просмотра изображений и рисунков. По умолчанию применяется значение, указанное в параметре Цвет изображения на странице Логические представления приложения OnDemand.

Просмотр преобразованных документов

Опция UseApplet файла AFP2HTML.INI является директивой для ODWEK, позволяющей применять для просмотра преобразованного вывода апплет AFP2HTML. Значение по умолчанию - TRUE. Если указано значение FALSE (апплет AFP2HTML не применяется для просмотра вывода), то вывод форматируется Web-браузером и отображается в его окне.

Как правило, для просмотра преобразованных документов IBM всегда рекомендует применять апплет AFP2HTML. Если документ сохранен на сервере OnDemand в качестве большого объекта, то апплет AFP2HTML добавляет удобные средства управления, позволяющие перемещаться по страницам документа.

Приложение F. Преобразование AFP в PDF

Служебная программа AFP2PDF Transform преобразует документы и ресурсы AFP в документы PDF. AFP2PDF Transform представляет служебную программу, предоставляемую IBM Printing Systems Division. Администратор устанавливает и настраивает AFP2PDF Transform на сервере HTTP. Более подробную информацию о служебной программе AFP2PDF Transform можно получить в представительстве IBM. Для документов AFP и ресурсов, обрабатываемых с помощью AFP2PDF Transform, необходимо указать опции конфигурации. В этом разделе приведены инструкции по указанию опций конфигурации.

Примечание: В данном документе AFP2PDF.INI является именем файла конфигурации. Для того чтобы указать файл, содержащий опции конфигурации, обратитесь к разделу “CONFIGFILE” на стр. 22.

Файл AFP2PDF.INI содержит опции конфигурации для AFP2PDF Transform. Как правило в файле AFP2PDF.INI указываются опции конфигурации для конкретных приложений AFP. Кроме того, вы можете указать набор опций по умолчанию. Для приложений AFP, не указанных в файле AFP2PDF.INI, в процессе преобразования документов и ресурсов программа AFP2PDF Transform применяет опции по умолчанию. Более подробная информация о процессе преобразования и соответствующих опциях приведена в документации по AFP2PDF Transform.

В следующих разделах приведена дополнительная информация о файле AFP2PDF.INI:

- Указание файла AFP2PDF.INI
- Просмотр преобразованных документов

Примечание: Для преобразования документов следует также указать параметр AFPVIEWING=PDF в разделе DEFAULT BROWSER (или в разделах других браузеров) файла ARSWWW.INI. Подробная информация приведена в разделе “AFPVIEWING” на стр. 31. (Если вы планируете работать с API Загрузить документ, укажите параметр _afr=PDF. Дополнительная информация приведена в разделе “Загрузить документ” на стр. 77.)

Указание файла AFP2PDF.INI

Ниже приведен пример файла AFP2PDF.INI:

```
[CREDIT-CREDIT]
OptionsFile=
ImageMapFile=creditImageMap.cfg
```

```
[default]
OptionsFile=
ImageMapFile=imagemap.cfg
AllObjects=0
```

Структура данного файла аналогична структуре файлов INI операционной системы Windows и содержит по одному разделу для каждого приложения AFP, а также один раздел по умолчанию. Заголовок раздела определяет группу приложений и приложение. Например, строка заголовка

```
[CREDIT-CREDIT]
```

Определяет группу приложений CREDIT и приложение CREDIT. Имена в строке заголовка разделяются с помощью символа – (дефис). Имена должны совпадать с именами групп приложений и приложений, определенными на сервере OnDemand. Если группа приложений содержит несколько приложений, то для каждого приложения следует создать отдельный раздел.

Параметры раздела [default] предназначены для обработки документов приложений AFP, не указанных в файле AFP2PDF.INI. Кроме того, параметры по умолчанию применяются, если в разделе приложения AFP не указана одна из опций.

Параметр OptionsFile определяет полный путь к файлу, в котором содержатся опции преобразования, применяемые AFP2PDF Transform. Опции преобразования необходимы для документов AFP, требующих специальной обработки. Подробная информация о файле опций преобразования приведена в документации по AFP2PDF Transform.

Параметр ImageMapFile задает файл преобразования изображений. Файл преобразования изображений позволяет удалять изображения из вывода, улучшать качество темных изображений, заменять существующие изображения на изображения, созданные AFP2PDF Transform. Преобразование изображений, часто встречающихся в документах AFP (например, логотипа компании), позволяет ускорить процесс преобразования документов. Если файл преобразования изображений указан, то он должен существовать и располагаться в каталоге, в котором содержатся программы AFP2PDF Transform. Инструкции по указанию каталога программ AFP2PDF Transform приведены в разделе “INSTALLDIR” на стр. 23. Подробная информация о файле преобразования изображений приведена в документации по AFP2PDF Transform.

Параметр AllObjects определяет способ обработки документов с помощью ODWEK, сохраняемых в виде больших объектов на сервере OnDemand. Значение по умолчанию - 0 (ноль). Оно означает, что ODWEK загружает только первый сегмент документа. Если указано значение 1, то перед отправкой документа клиенту ODWEK загружает и преобразует все сегменты документа. **Примечание:** Применение поддержки больших документов может привести к значительным задержкам при просмотре очень больших документов на клиенте.

Просмотр преобразованных документов

Для просмотра преобразованных документов необходимо получить программу Adobe Acrobat для браузеров, применяемых в вашей организации.

Приложение G. файлы конфигурации сервера HTTP

В этом разделе приведены примеры следующих файлов конфигурации сервера HTTP:

- Исходный сервер HTTP (больше не поддерживается)
- HTTP Apache Server
- Сервер приложений WebSphere

Примечание: Дополнительная информация по настройке приведена в документации HTTP.

HTTP Apache Server

Примечания:

1. В строке Получение запросов замените строку **n.n.n.n** на IP-адрес системы iSeries, в которой установлено программное обеспечение OnDemand.
2. В строке Получение запросов замените **p** на номер порта TCP/IP, применяемого для обмена данными между клиентом и сервером HTTP. Номер порта TCP/IP по умолчанию - 80.
3. В строке DefaultNetCCSID замените строку **xxxx** на CCSID экземпляра OnDemand.
4. В строке Имя сервера замените **f.q.h.n** на полное имя хоста TCP/IP системы iSeries, в которой установлено программное обеспечение OnDemand.
5. Для языков DBCS измените строку CGIConvMode на следующую строку:

```
CGIConvMode %%EBCDIC_JCD/MIXED%%
```

```
#HTTP Apache Server Configuration for the ODWAPACHE server
```

```
Listen n.n.n.n:p
```

```
DefaultNetCCSID xxxx
```

```
HostNameLookups off
```

```
RuleCaseSense OFF
```

```
LimitRequestBody 102400
```

```
ErrorLog /www/odwapache/logs/error_log
```

```
ServerName f.q.h.n
```

```
DocumentRoot /www/odwapache
```

```
UseCanonicalName Off
```

```
CGIConvMode %%EBCDIC/MIXED%%
```

```
ScriptLogLevel 200
```

```
ScriptLog /www/odwapache/logs/cgi_log
```

```
Alias /logon /www/odwapache/odw_logon.htm
```

```
AliasMatch ^/images/(.*)$ /QIBM/UserData/OnDemand/www/images/$1
```

```
AliasMatch ^/applets/com/ibm/edms/od/(.*)$ /QIBM/ProdData/OnDemand/www/applets/$1
```

```
AliasMatch ^/applets/(.*)$ /QIBM/ProdData/OnDemand/www/applets/$1
```

```
ScriptAliasMatch ^/scripts/arswww.cgi$ /QSYS.LIB/QRDARS.LIB/ARS3WCGI.PGM
```

```
ScriptAliasMatch ^/scripts/arswww\.cgi/(.*)$ /QSYS.LIB/QRDARS.LIB/ARS3WCGI.PGM
```

```
AddType www/unknown cab
```

```
AddType www/unknown jar
```

```
AlwaysDirectoryIndex On
```

```
DirectoryIndex index.html
```

```
<Directory/>
```

```
Options None
```

```
Options +ExecCGI
```

```
AllowOverride None
```

```
Order deny,allow
```

```
Deny from all
```

```
<LimitExcept GET HEAD OPTIONS TRACE POST PUT>
```

```
</LimitExcept>
```

```
</Directory>
```

```
<Directory /QIBM/ProdData/OnDemand/www/applets/>
```

```
Order allow,deny
Allow from All
</Directory>
<Directory /QIBM/ProdData/OnDemand/www/images/>
Order allow,deny
Allow from All
Options +FollowSymLinks
</Directory>
<Directory /QIBM/ProdData/OnDemand/www/samples/>
Order allow,deny
Allow from All
Options +FollowSymLinks
</Directory>
<Directory /www/odwapache/>
Order allow,deny
Allow from All
</Directory>
<Directory /QSYS.LIB/QRDARS.LIB>
Order allow,deny
Allow from All
Options +ExecCGI
</Directory>
```

Сервер приложений WebSphere

Пример файла конфигурации WebSphere можно загрузить с Web-страницы поддержки IBM Content Manager OnDemand for iSeries, расположенной по адресу <http://www.ibm.com/software/data/ondemand/400/support.html>. *Пример ODWEK WebSphere example* приведен в категории Technotes.

Приложение Н. Не создавать вывод HTML

Директива `_nohtml` позволяет определить тип вывода, создаваемого функциями (такими как `Вход в систему`). По умолчанию ODWEK создает вывод HTML. Если указано значение `_nohtml=1`, то ODWEK создает вывод ASCII с ограничителями. В этом разделе описан вывод ASCII с ограничителями, создаваемый программным обеспечением ODWEK.

Вывод ASCII с ограничителями

Вывод ASCII с ограничителями, создаваемый программным обеспечением ODWEK, представляет собой набор записей вывода, содержащий символьные значения, ключевые слова, функцию, запись, а также ограничители и разделители:

- Символьные значения представляют собой данные вывода функции, отличающиеся от ключевых слов, ограничителей и разделителей. Например, символьными значениями являются такие данные, как имя функции, вызываемой следующей, имя папки, имена полей папки, операторы поиска и значения полей.
- Ключевые слова состоят из конкретных строк символов. Например, ключевыми словами являются такие строки, как `ACTION`, `DOC`, `FOLDER`, `NUMROWS` и `ROW`.
- Ограничители функции состоят из строк символов `[BEGIN]` и `[END]`.
- Разделителем записи является символ новой строки `\n`. Символом новой строки ограничиваются все записи.
- По умолчанию в качестве ограничителей и разделителей строк применяются символ `(\)`, а также символы `([` и `])`. Например:
`[имя-папки\описание-папки]`

Если ключевое слово состоит из нескольких строковых значений, то эти значения разделяются с помощью символа `\`. Набор строковых значений каждого ключевого слова ограничивается символами правой и левой прямых скобок.

Некоторые строковые значения можно сохранить в виде списка, разделенного с помощью символов `\` и заключенного в прямые скобки. Например, список допустимых операторов поиска может выглядеть следующим образом:

```
[1\2\4\8\16\32]
```

Вы можете переопределить символы, применяемые в качестве ограничителей и разделителей. Более подробная информация приведена в разделе “[NO HTML]” на стр. 29.

- Отдельная пустая строка обозначается с помощью двух символов двойных кавычек (`""`). Пустой список обозначается с помощью двух прямых скобок (`[]`).

Вход в систему

Ниже приведен пример вывода ASCII с ограничителями, созданного функцией `Вход в систему`:

```
[BEGIN]\nACTION=searchCriteriaUrl\nFOLDER=[folderName\folderDesc]\nFOLDER=[folderName\folderDesc]\n
```

⋮
[END]\n

Примечания

1. Строка `searchCriteriaUrl` задает имя и параметры следующей выполняемой функции.
2. Строка `folderName` задает имя папки. Имя не заключено в кавычки.
3. Строка `folderDesc` задает описание папки. Описание не заключено в кавычки.

Критерии поиска

Ниже приведен пример вывода ASCII с ограничителями, созданного функцией Критерии поиска:

```
[BEGIN]\nACTION=hitListUrl\nDISPLAY_ORDER=[field1^field2^...fieldN]\nNUMROWS=numberOfRows\nROW=[criteriaName^[validOp]^defOp]^inpType^inpAssocData]\n
```

⋮
[END]\n

Примечания

1. Строка `hitListUrl` задает имя и параметры следующей выполняемой функции.
2. Ключевое слово `DISPLAY_ORDER` задает порядок отображения файлов папки.
3. Строка `numberOfRows` задает число записей ключевого слова `ROW`. Функция создает запись ключевого слова `ROW` для каждого поля поиска.
4. Строка `criteriaName` представляет критерий поиска для поля поиска. Критерий поиска не заключается в кавычки.
5. Строка `validOp` задает список целых значений, представляющий допустимые операторы поиска для поля поиска:

1	Равно
2	Не равно
4	Меньше
8	Меньше или равно
16	Больше
32	Больше или равно
64	Входит в состав
128	Не входит в состав
256	Похоже на
512	Не похоже на
1024	Лежит в диапазоне
2048	Не лежит в диапазоне
6. Строка `defOp` задает целое значение, представляющее оператор поиска по умолчанию.
7. Строка `inpType` представляет тип поля поиска:

A	Поиск в тексте комментариев
C	Выбор
N	Нормальный
S	Фрагмент

- T** Текстовый поиск
 - Z** Поиск цвета комментариев
8. Строка `inpAssocData` представляет собой список, связанный с `defOp` и `inpType`. См. Табл. 15.

Таблица 15. Оператор по умолчанию и тип ввода, связанные с `inpAssocData`

defOp	inpType	inpAssocData
Between, Not Between	N	Пустое значение: [] или список: [defaultField1^...^defaultFieldN] Например: ["01/31/96"^"01/31/97"] ["01/31/96"^" ["^"01/31/97"]
Другие допустимые операторы	A, N, T, Z	Пустое значение: [] или строковое значение, представляющее значение поля по умолчанию
Другие допустимые операторы	C, S	[[listOfChoices]^defaultChoice] Например: [["JFIF"^"TIFF"^"PCX"]^"TIFF"] [["JFIF"^"TIFF"^"PCX"]^"]

Список найденных документов

Ниже приведен пример вывода ASCII с ограничителями, созданного функцией Список найденных документов:

```
[BEGIN]\n
ACTION=hitListURL\n
MSG=Only 20 documents can be listed for this folder.
DOC=[criteria1^criteria2^criteriaN^docid^fileType^docLocation]\n
:
[END]\n
```

Примечания

1. Строка `hitListURL` задает имя и параметры следующей выполняемой функции.
2. Ключевое слово `MSG` представляет собой пример сообщения об ошибке в выводе ASCII с ограничителями. По умолчанию ODWEK отправляет клиентам сообщения об ошибках. Однако, если функция содержит директиву `_nohtml=1`, то вместо этого в выводе ASCII с ограничителями создается текст сообщения.
3. Строки `criteria1`, `criteria2` и `criteriaN` представляют значения критериев поиска. Значения перечислены в порядке, в котором они отображаются в списке документов. Значения не заключены в кавычки.
4. Строка `docid` задает идентификатор документа.
5. Строка `fileType` задает тип данных документа:
 - A** AFP
 - B** BMP
 - E** Email
 - F** JFIF
 - G** GIF
 - L** Line
 - N** Нет
 - O** Определяется OD

P	PDF
T	TIFF
U	Определяется пользователем
X	PCX

6. Строка `docLocation` задает расположение документа:

0	Неизвестно
1	Кэш-память OnDemand
2	Архив
3	Внешняя кэш-память

Просмотреть комментарии

Ниже приведен пример вывода ASCII с ограничителями, созданного функцией
Просмотреть комментарии:

```
[BEGIN]\nNOTE 4: 15:42:44 PM Mountain Standard Time Thursday November 19, 1998...\nPublic - Cannot be copied to another server\nTest note from the OnDemand Internet Client.\n[END]\n
```

Сообщение об ошибке

Ниже приведен пример вывода ASCII с ограничителями, создаваемого в случае
возникновения ошибки:

```
[ERROR]\nID=nnnn\nMSG=errorMessageText\n
```

Примечания

1. Строка `nnnn` является номером сообщения об ошибке.
2. Строка `errorMessageText` задает текст сообщения об ошибке.

Приложение I. Поддержка языков

В этом разделе приведена информация, которая может помочь администраторам в настройке ODWEK для работы с языками DBCS.

Параметры CODEPAGE и LANGUAGE в файле ARSWWW.INI позволяют указать опции настройки языков (NL).

Параметр CODEPAGE определяет кодовую страницу сервера ODWEK и должен быть совместим с базой данных OnDemand сервера библиотеки OnDemand. Параметр CODEPAGE следует указывать **только** в том случае, если кодовая страница рабочей станции, в которой выполняется приложение ODWEK, отличается от кодовой страницы базы данных OnDemand сервера библиотеки OnDemand. Система применяет кодовую страницу рабочей станции, в которой выполняется приложение ODWEK, в качестве значения по умолчанию.

Параметр LANGUAGE определяет каталог сообщений, применяемый ODWEK для отображения сообщений.

В разделе Табл. 16 приведен список кодовых страниц DBCS и языков, поддерживаемых OnDemand. В столбце **CODEPAGE=** приведен список значений кодовой страницы, которые следует указывать **только** в том случае, если кодовая страница рабочей станции, в которой выполняется приложение ODWEK, отличается от кодовой страницы базы данных OnDemand. В столбце **LANGUAGE=** приведены значения, связанные с каталогами переведенных сообщений.

Примечание: Операционная система Linux не указана в таблице, так как она не может быть целевым сервером приложения ODWEK. Хотя приложение ODWEK может выполняться в операционной системе Linux, она не поддерживается базой данных OnDemand (сервером библиотеки OnDemand).

Таблица 16. Кодовые страницы DBCS, языки, наборы символов и локали

Регион	LANGUAGE=	OS	Кодовая страница базы данных	CODEPAGE=	Набор символов	Локаль
Китай (PRC)	CHS	AIX	1383	1383	IBM_eucCN	zh_CN
		HP-UX	1383	1383	hp15CN	zh_CN. hp15CN
		Solaris	1383	1383	gb2312	zh
		Windows	1386	1386	GBK	—
		z/OS или OS/390 (EBCDIC)	935	935	IBM-935	—

Таблица 16. Кодовые страницы DBCS, языки, наборы символов и локали (продолжение)

Регион	LANGUAGE=	OS	Кодовая страница базы данных	CODEPAGE=	Набор символов	Локаль
Япония	JPN	AIX	954	954	IBM_eucJP	ja_JP
		HP-UX	954	954	eucJP	ja_JP.eucJP
		Solaris	954	954	eucJP	ja
		Windows	943	943	IBM-943	—
		z/OS или OS/390 (EBCDIC)	939	939	IBM-939	—
Южная Корея	KOR	AIX	970	970	IBM_eucKR	ko_KR
		HP-UX	970	970	eucKR	ko_KR.eucKR
		Solaris	970	970	5601	ko
		Windows	1363	1363	1363	—
		z/OS или OS/390 (EBCDIC)	933	933	IBM-933	—
Тайвань	СНТ	AIX	964	964	IBM_eucTW	zh_TW
		HP-UX	964	964	eucTW	zh_TW.eucTW
		Solaris	964	964	cns11643	zh_TW
		Windows	950	950	big5	—
		z/OS или OS/390 (EBCDIC)	937	937	IBM-937	—

Дополнительная информация о настройке языков DBCS в системе OnDemand приведена в разделе “Поддержка национальных языков” книги *Планирование и установка*.

Приложение J. Инструменты определения неполадок

Для сбора информации о системе и документах воспользуйтесь инструментами, перечисленными в разделе Табл. 17. Данная информация предназначена для устранения неполадок, связанных с настройкой ODWEK, а также для поддержки пользователей, у которых возникли неполадки при работе с апплетами и встраиваемыми модулями.

Таблица 17. Инструменты определения неполадок

Инструмент	Назначение	Инструкции по применению
Вывод HTML	Сохраняет копию текста на языке HTML, возвращаемого ODWEK браузеру клиента.	Выберите в меню Файл браузера пункт Сохранить как.
Файлы протоколов сервера	Сохраняется информация о входе на сервер, информация об ошибках, а также информация о сервере.	<p>Выполните следующие действия:</p> <ol style="list-style-type: none"> 1. В разделе DEBUG файла ARSWWW.INI укажите в параметре LOG значение 1. ODWEK создает файл протокола ARSWWW.LOG и записывает его в каталог, указанный в параметре LOGDIR. (Каталог по умолчанию - /QIBM/UserData/OnDemand/WWW/LOG .) <p>Важная информация: Раздел DEBUG должен быть первым исполняемым оператором файла ARSWWW.INI.</p> <ol style="list-style-type: none"> 2. Настройка функций ведения протокола на сервере HTTP. (Процесс настройки протокола, протоколы и опции сбора информации зависят от сервера HTTP.) <p>Примечание: IBM рекомендует разрешать занесение информации в протокол только в тех случаях, когда это необходимо, например, для восстановления неполадки, так как при этом в файл протокола записываются значительные объемы информации. Если вам необходимо разрешить ведение протокола на длительный период времени, убедитесь, что файл протокола расположен на устройстве, в котором доступно достаточно памяти. Рекомендуется периодически удалять старые файлы протоколов.</p>

Таблица 17. Инструменты определения неполадок (продолжение)

Инструмент	Назначение	Инструкции по применению
Консоль Java	Отображает сообщения, созданные апплетами.	<ul style="list-style-type: none"> • Netscape: В меню Communicator выберите пункт Инструменты, затем Консоль Java. • Internet Explorer: <ol style="list-style-type: none"> 1. В меню Вид выберите пункт Опции Internet. 2. На странице Дополнительно выберите Консоль Java. 3. Перезапустите браузер. 4. В меню Вид выберите Консоль Java.
Средство трассировки AFP Web Viewer	Сбор подробной информации о документах AFP, просматриваемых с помощью AFP Web Viewer.	<p>Убедитесь, что в файле FLDPORT2.INI рабочей станции существуют следующие разделы:</p> <pre>[Misc] ViewTraceFile=d:\temp\afpplogin.log Trace=TRUE</pre> <p>Проверьте путь к файлу протокола. Не забудьте отключить ведение протокола после сбора необходимой информации.</p>
Системный журнал OnDemand	Сохранение системных сообщений (таких как вход в систему и выход из нее), сообщений группы приложений, связанных с документами (таких как запрос и загрузка) и комментариями.	<p>Выполните следующие действия:</p> <ol style="list-style-type: none"> 1. Разрешите ведение системного журнала и группы приложений на сервере OnDemand. Обновите системные параметры сервера с помощью клиента администрирования. 2. Разрешите применение сообщений группы приложений для занесения в протокол. Обновите опции занесения сообщений в протокол для группы приложений с помощью клиента администрирования.

Примечания

Настоящая документация была разработана для продуктов и услуг, предлагаемых на территории США.

IBM может не предлагать продукты и услуги, упомянутые в этом документе, в других странах. За дополнительной информацией о доступных продуктах и услугах обратитесь в местное представительство компании IBM. Ссылки на продукты, программы и услуги компании IBM не означают, что применять можно только эти программы, продукты или услуги. Вместо них можно использовать продукты, программы и услуги, не являющиеся интеллектуальной собственностью компании IBM. Однако в этом случае ответственность за проверку работы этих продуктов, программ и услуг возлагается на пользователя.

IBM могут принадлежать патенты или заявки на патенты, относящиеся к материалам этого документа. Предоставление вам настоящего документа не означает предоставления каких-либо лицензий на эти патенты. Запросы на приобретение лицензий можно отправлять по следующему адресу:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

Запросы на лицензии, связанные с информацией DBCS, следует направлять в отдел интеллектуальной собственности в местном представительстве IBM или в письменном виде по следующему адресу:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

Следующий абзац не относится к Великобритании, а также к другим странам, в которых это заявление противоречит местному законодательству:
КОМПАНИЯ INTERNATIONAL BUSINESS MACHINES CORPORATION ПРЕДОСТАВЛЯЕТ НАСТОЯЩУЮ ПУБЛИКАЦИЮ НА УСЛОВИЯХ "КАК ЕСТЬ", БЕЗ КАКИХ-ЛИБО ЯВНЫХ ИЛИ ПОДРАЗУМЕВАЕМЫХ ГАРАНТИЙ, ВКЛЮЧАЯ, НО НЕ ОГРАНИЧИВАЯСЬ ЭТИМ, НЕЯВНЫЕ ГАРАНТИИ СОБЛЮДЕНИЯ ПРАВ, КОММЕРЧЕСКОЙ ЦЕННОСТИ И ПРИГОДНОСТИ ДЛЯ КАКОЙ-ЛИБО ЦЕЛИ.

В некоторых странах запрещается отказ от каких-либо явных и подразумеваемых гарантий при заключении определенных договоров, поэтому данное заявление может не действовать в вашем случае.

В данной публикации могут встретиться технические неточности и типографские опечатки. В информацию периодически вносятся изменения, которые будут учтены во всех последующих изданиях настоящей публикации. Компания IBM может вносить изменения в продукты и программы, описанные в этой публикации в любое время без предварительного уведомления.

Все встречающиеся в данной документации ссылки на Web-сайты других компаний предоставлены исключительно для удобства пользователей и не являются рекламой этих Web-сайтов. Материалы, размещенные на этих Web-сайтах, не являются частью информации по данному продукту IBM и ответственность за применение этих материалов лежит на пользователе.

IBM может использовать и распространять любую предоставленную вами информацию на свое усмотрение без каких-либо обязательств перед вами.

Для получения информации об этой программе для обеспечения: (i) обмена информацией между независимо созданными программами и другими программами (включая данную) и (ii) взаимного использования информации, полученной в ходе обмена, пользователи данной программы могут обращаться по адресу:

IBM Corporation
Software Interoperability Coordinator
3605 Highway 52 N
Rochester, MN 55901–7829
U.S.A.

Такая информация может предоставляться на определенных условиях, включая, в некоторых случаях, уплату вознаграждения.

Лицензионные программы, описанные в этом документе, и все доступные лицензионные материалы предоставляются компанией IBM в рамках Соглашения с заказчиком IBM, Международного соглашения о лицензии на программу IBM или аналогичного соглашения между сторонами.

Информация о продуктах других изготовителей получена от поставщиков этих продуктов, из их официальных сообщений и других общедоступных источников. IBM не выполняла тестирование этих продуктов других фирм и не может подтвердить точность заявленной информации об их производительности, совместимости и других свойствах. Запросы на получение дополнительной информации об этих продуктах должны направляться их поставщикам.

Лицензия на авторские права:

Эта информация содержит примеры прикладных программ на исходном языке, которые позволяют объяснить технику программирования для различных платформ. Вы можете копировать, изменять и распространять эти программы бесплатно для разработки, применения, сбыта и распространения прикладных программ, соответствующих интерфейсу прикладного программирования платформы, для которой они предназначены. Они не проверялись для работы во всех условиях. Поэтому компания IBM не может дать гарантию, как явную, так и подразумеваемую, надежности, удобства обслуживания и стабильной работы этих программ. Вы можете копировать, изменять и распространять эти программы бесплатно для разработки, применения, сбыта и распространения прикладных программ, соответствующих интерфейсам прикладного программирования IBM.

Если данный документ представлен в электронном виде, в нем могут отсутствовать фотографии и цветные иллюстрации.

Товарные знаки

Advanced Function Presentation, AFP, AS/400, IBM, iSeries, Operating System/400, OS/390, OS/400, Redbooks, WebSphere и z/OS являются торговыми марками International Business Machines Corporation в США и других странах.

Java и все торговые марки Java являются торговыми марками Sun Microsystems Inc.

Lotus является торговой маркой Lotus Development Corporation в США и других странах.

Microsoft, Windows и Windows NT являются зарегистрированными торговыми марками Microsoft Corporation в США и других странах.

Названия других компаний продуктов и услуг могут быть товарными или служебными знаками других компаний.

Индекс

A

- AFP Web viewer
 - настройка 51
- AFP Web Viewer
 - настройка установки 47
 - о программе 1
 - пользовательские файлы 47
 - преобразование шрифтов AFP 51
 - требования 46
 - установка 45
 - установка пользовательских файлов 47
 - шрифты 51
 - шрифты AFP 51
- AFP2PDF Transform
 - настройка 133
 - разрешение применения 22
 - установка 10
- AFP2WEB Transform
 - настройка 129
 - разрешение применения 20
 - установка 10
- API
 - вход в систему 71
 - выход из системы 69
 - добавить комментарий 60
 - загрузить документ 77
 - изменить пароль 63
 - информация диагностики 94
 - классы 91
 - комментарии 60, 85
 - критерии поиска 80
 - обновить документ 83
 - обработка исключительных ситуаций 94
 - печатать документ 73
 - пример исходного кода 91
 - примеры 91
 - просмотреть комментарии 85
 - руководство программиста 91
 - руководство программиста API Java 91
 - сервер, печать 73
 - система, среда 92
 - список найденных документов 65
 - справочник 59, 87, 89
 - справочник по API CGI 59
 - справочник по API Java 89
 - среда, переменные 92
 - структура 91
 - трассировка и информация диагностики 94
- API Java
 - о программе 1
 - руководство программиста 91
 - справочник 89
 - требования к программному обеспечению 1
- ARSWWW.INI file
 - раздел @SRV@_server 12

C

- CGI API
 - справочник 59

- cookie 33, 46
- CREDIT.HTM 41

D

- DBCS 141

I

- Image Web viewer
 - настройка 52
- Image Web Viewer
 - о программе 1
 - требования 46
 - установка 45

J

- JVM 46

L

- LOGON.HTM 41

N

- NLS 16, 18, 141

O

- ODCallback 118
- ODCriteria
 - документы, обновление 124
 - имя 100
 - обновление документа 124
 - операнды 100, 107, 109
 - поисковые значения 100, 107, 109
- ODCriteria.getFixedValues 109
- ODCriteria.getName 100
- ODCriteria.getOperand 100, 107
- ODCriteria.getType 109
- ODCriteria.getValidOperands 109
- ODCriteria.getValues 109
- ODCriteria.setOperand 109, 124
- ODCriteria.setSearchValue 100, 124
- ODCriteria.setSearchValues 100, 107, 109
- ODFolder
 - группы приложений 99
 - документ, загрузка 116
 - документ, печать 119
 - загрузить документ 116
 - заккрытие 99, 100, 105, 107
 - имя 100, 114
 - критерии 100, 107, 109
 - описание 100
 - отмена поиска 107
 - печать документов 119
 - поиск 100, 105, 107, 109, 114, 116

ODFolder (продолжение)
 порядок отображения 100, 114
 сообщение 100
 ODFolder.close 99, 100, 105, 107, 116
 ODFolder.getApplGroups 99
 ODFolder.getCriteria 100, 107, 109
 ODFolder.getDescription 100
 ODFolder.getDisplayOrder 100, 105, 114
 ODFolder.getName 100, 114
 ODFolder.getNumApplGroups 99
 ODFolder.getSearchMessage 100
 ODFolder.printDocs 119
 ODFolder.retrieve 116
 ODFolder.search 100, 105, 107, 114, 116
 ODFolder.setApplGroupForSearchWithSQL 105
 ODHHit
 документ, загрузка 116
 документ, обновление 124
 загрузить документ 116
 заметки 121, 123
 комментарии 121, 123
 обновление документов 124
 отображаемые значения 100, 105
 показать значение 114
 расположение документа 100
 список документов 114
 тип данных MIME 100
 тип документа 100
 ODHHit.addNote 123
 ODHHit.getDisplayValue 100, 105, 114, 124
 ODHHit.getDisplayValues 100
 ODHHit.getDocId 100, 116
 ODHHit.getDocLocation 100
 ODHHit.getDocType 100
 ODHHit.getMimeType 100
 ODHHit.getNotes 121, 123
 ODHHit.retrieve 116
 ODHHit.update 124
 ODNNote
 время 121
 дата 121
 ИД пользователя 121
 имя группы 121
 комментарии 121, 123
 расположение 121
 страница 121
 текст 121
 цвет 121
 ODNNote.getColor 121
 ODNNote.getDateTime 121
 ODNNote.getGroupName 121
 ODNNote.getOffsetX 121
 ODNNote.getOffsetY 121
 ODNNote.getPageNum 121
 ODNNote.getText 121
 ODNNote.getUserid 121
 ODNNote.isOkToCopy 121, 123
 ODNNote.isPublic 121, 123
 ODNNote.setGroupName 123
 ODNNote.setText 123
 ODServer
 документ, загрузка 116
 загрузить документ 116
 задание пароля 127
 ИД пользователя 96, 97
 изменение паролей 127
 имя папки 112

ODServer (продолжение)
 имя приложения 97
 локальный каталог 97
 описание папки 112
 отключение от 97
 открытие папки 109
 открыть папку 116
 отмена поиска 107
 папка, открытие 116
 пароль 96, 97, 127
 подключение к 97
 подключение к серверу 96
 порт 97
 принтеры 119
 принтеры сервера 119
 сервер 96, 97
 тип соединения 97
 установка и просмотр ИД пользователей 96
 установка и просмотр паролей 96
 ODServer.cancel 107
 ODServer.changePassword 127
 ODServer.getConnectType 97
 ODServer.getFolderNames 112
 ODServer.getFoldersDescription 112
 ODServer.getLocalDir 97
 ODServer.getNumFolders 112
 ODServer.getPassword 96, 97
 ODServer.getPort 97
 ODServer.getServerName 96, 97
 ODServer.getServerPrinters 119
 ODServer.getUserId 96, 97
 ODServer.logoff 96, 97
 ODServer.logon 96, 97
 ODServer.openFolder 109, 116
 ODServer.retrieve 116
 ODServer.setApplicationName 97
 ODServer.setConnectType 97
 ODServer.setLocalDir 97
 ODServer.setPassword 96, 97
 ODServer.setPort 97
 ODServer.setServer 96, 97
 ODServer.setUserId 96, 97
 ODServer.terminate 96, 97
 OS/400
 установка 10

T

TEMPLATE.HTM 42

W

Web-приложения
 примеры 41
 Web-страницы
 примеры 41

A

аннотации 36
 апплет AFP2HTML Java
 о программе 1, 5
 параметр APPLETCACHEDIR 13
 поддержка больших объектов 21, 131
 требования 10, 46
 установка 45

- апплет AFP2PDF Java
 - каталог 23
 - разрешение применения 23
- апплет Строковые данные Java
 - о программе 4
- апплет строковых данных Java
 - о программе 1
 - параметр APPLETCACHEDIR 13
 - требования 46
 - установка 45
- апплеты 13
 - каталог 23
 - о программе 1
 - параметр APPLETCACHEDIR 13
 - поддержка больших объектов 21, 131
 - разрешение применения 23
 - требования 46
 - установка 45
- апплеты Java
 - о программе 1
 - установка 45
- Апплеты Java
 - каталог 23
 - о программе 4, 5
 - параметр APPLETCACHEDIR 13
 - поддержка больших объектов 21, 131
 - разрешение применения 23
 - требования 46
- атрибут метода тега form 8

Б

- большие объекты 21, 131
- браузеры
 - cookie 46
 - JVM 46
 - Виртуальная машина Java 46
 - поддержка 46

В

- ведение протокола 38, 143
- Виртуальная машина Java 46
- вложения 27, 28
- вложения BMP 28
- вложения GIF 28
- вложения TXT 28
- временная память 19
- временный рабочий каталог 19
- встраиваемые модули
 - о программе 1
 - установка 45
- вход в систему
 - API 71
 - вывод ASCII с ограничителями 137
 - описание функции 6
 - параметры 71
 - пример вызова функции 72
- вывод ASCII
 - вход в систему 137
 - комментарии 140
 - критерии поиска 138
 - просмотреть комментарии 140
 - созданный с помощью OnDemand 137
 - сообщение об ошибке 140
 - сообщения 140

- вывод ASCII (*продолжение*)
 - список найденных документов 139
 - формат 137
- вывод ASCII с ограничителями
 - вход в систему 137
 - комментарии 140
 - критерии поиска 138
 - ограничители 29
 - просмотреть комментарии 140
 - созданный с помощью OnDemand 137
 - сообщение об ошибке 140
 - сообщения 140
 - список найденных документов 139
 - формат 137
- вывод без HTML 29, 137
- выход из системы
 - API 69
 - описание функции 6
 - параметры 69
 - пример вызова функции 70

Г

- группы приложений в папке
 - API Java 99

Д

- добавить комментарий
 - API 60
 - описание функции 6
 - параметры 60
 - пример вызова функции 62
- документы
 - AFP 31
 - EMAIL 32
 - загрузка 32
 - кэш-память 14
 - обновление с помощью API Java 124
 - печать с помощью API Java 119
 - преобразование 31, 32, 34
 - просмотр 31, 32, 34
 - ссылки 35
 - строковые данные 34
 - тип данных MIME 23
 - тип носителя 23
- документы AFP
 - преобразование 31
 - просмотр 31
 - тип данных MIME 25
 - тип носителя 25
- документы BMP
 - тип данных MIME 25
 - тип носителя 25
- документы EMAIL
 - преобразование 32
 - просмотр 32
 - тип данных MIME 25
 - тип носителя 25
- документы GIF
 - тип данных MIME 25
 - тип носителя 25
- документы JFIF
 - тип данных MIME 26
 - тип носителя 26

- документы PCX
 - тип данных MIME 26
 - тип носителя 26
- документы PDF
 - тип данных MIME 27
 - тип носителя 27
- документы TIFF
 - тип данных MIME 27
 - тип носителя 27
- документы строковых данных
 - преобразование 34
 - просмотр 34
 - тип данных MIME 26
 - тип носителя 26
- документы, Java API 114, 116

З

- загрузить документ
 - API 77
 - описание функции 6
 - параметры 77
 - пример вызова функции 79
- загрузка
 - документы 32
- загрузка документа 116
- задание пароля 127
- заметки 31, 36
- защита 7, 19, 33
- защита данных 7
- защита сервера 7, 19
- значения поиска, API Java 105

И

- ИД пользователей
 - API Java 96, 97
 - кэш-память 16
- изменение паролей 127
- изменить пароль
 - API 63
 - описание функции 6
 - параметры 63
 - пример вызова функции 64
- имя папки, API Java 112
- имя приложения
 - API Java 97
- имя хоста 12
- информация диагностики 94

К

- каталог изображений 17
- каталог кэш-памяти 14
- классы 91
- кодовая страница 16, 141
- комментарии 31
 - API 60, 85
 - API Java 121, 123
 - вывод ASCII с ограничителями 140
 - описание функции 6, 7
 - параметры 60, 85
 - пример вызова функции 62, 86
- критерии поиска
 - API 80
 - API Java 100, 105, 109

- критерии поиска (*продолжение*)
 - вывод ASCII с ограничителями 138
 - описание функции 7
 - параметры 80
 - пример вызова функции 82
 - строка SQL 105
- кэш-память 14, 15, 16
- кэширование документов 14

Л

- локальный каталог
 - API Java 97

М

- максимальное число совпадений 34
- метод GET 8
- метод POST 8

Н

- настройка
 - программное обеспечение ODWEK 9
 - сервер HTTP 9
 - файл ARSWWW.INI 11
 - файл конфигурации AFP2HTML 129
 - файл конфигурации AFP2PDF 133
- номер порта 11, 12

О

- о данной публикации 1
- о соединении Internet OnDemand 1
- обзор 1
- обновить документ
 - API 83
 - API Java 124
 - описание функции 7
 - параметры 83
 - пример вызова функции 84
- обработка исключительных ситуаций 94
- ограничители 29
- ограничители вывода 29
- операнды, API Java 100
- Описание папки, API Java 112
- определение неполадок 143
- опции Web-сервера
 - DBCS 141
 - NLS 141
 - кодовая страница 141
 - язык 141
- опции браузера
 - раздел browser 36
 - раздел DEFAULT BROWSER 30
- опции браузера по умолчанию 30
- опции сервера HTTP
 - AFP2PDF Transform 22
 - AFP2WEB Transform 20
 - debug 37
 - браузер по умолчанию 30
 - браузеры 30, 36
 - параметр APPLETDIR 13
 - параметр BEGIN 29
 - параметр CACHEDIR 14

- опции сервера HTTP *(продолжение)*
 - параметр CACHEDOCS 14
 - параметр CACHEMAXTHRESHOLD 15
 - параметр CACHEMINTHRESHOLD 15
 - параметр CACHESIZE 15
 - параметр CACHEUSERIDS 16
 - параметр CODEPAGE 16
 - параметр CONFIGFILE 21, 22
 - параметр END 29
 - параметр IMAGEDIR 17
 - параметр INSTALLDIR 21, 23
 - параметр LANGUAGE 18
 - параметр REPORTSERVERTIMEOUT 20
 - параметр SEPARATOR 29
 - параметр SERVERACCESS 20
 - параметр ShowSearchString 18
 - параметр TEMPDIR 19
 - параметр TEMPLATEDIR 19
 - параметр USEEXECUTABLE 21, 23
 - раздел AFP2HTML 20
 - раздел AFP2PDF 22
 - раздел ATTACHMENT IMAGES 27
 - раздел CONFIGURATION 13
 - раздел MIMETYPES 23
 - раздел NOHTML 29
 - раздел SECURITY 19
- опции сервера OnDemand
 - значения по умолчанию 11
 - параметр HOST 12
 - параметр PORT 11, 12
 - параметр PROTOCOL 12, 13
 - параметры 12
 - раздел @SRV@_DEFAULT 11
 - раздел @SRV@_server 12
- отключение от сервера 97
- отмена поиска 107
- отображаемые значения, API Java 100
- ошибки 94, 143

П

- папка, поиск с API Java 100, 105, 107, 109, 114
- папка, просмотр списка групп с API Java 99
- параметр ADDEXTENSION 30
- параметр ADDFIELDSTODOCID 30
- параметр ADDNOTES 31
- параметр AFPVIEWING 31
- параметр APPLETCACHEDIR 13
- параметр APPLETDIR 13
- параметр AUTODOCRETRIEVAL 32
- параметр BEGIN 29
- параметр CACHEDIR 14
- параметр CACHEDOCS 14
- параметр CACHEMAXTHRESHOLD 15
- параметр CACHEMINTHRESHOLD 15
- параметр CACHESIZE 15
- параметр CACHEUSERIDS 16
- параметр CODEPAGE 16, 141
- параметр CONFIGFILE 21, 22
- параметр DOCSIZE 16
- параметр EMAILVIEWING 32
- параметр ENCRYPTCOOKIES 33
- параметр ENCRYPTURL 33
- параметр END 29
- параметр FOLDERDESC 33
- параметр HOST 12
- параметр IMAGEDIR 17

- параметр INSTALLDIR 21, 23
- параметр LANGUAGE 18, 141
- параметр LINEVIEWING 34
- параметр LOG 38, 143
- параметр LOGDIR 38
- параметр MAXHITS 34
- параметр NOLINKS 35
- параметр ODApplet.jre.path.IE 35
- параметр ODApplet.jre.path.NN 35
- параметр ODApplet.jre.version 35
- параметр ODApplet.version 35
- параметр PORT 11, 12
- параметр PROTOCOL 12, 13
- параметр REPORTSERVERTIMEOUT 20
- параметр SEPARATOR 29
- параметр SERVERACCESS 20
- параметр SERVERPRINT 35
- параметр SERVERPRINTERS 36
- параметр SHOWDOCLOCATION 36
- параметр ShowSearchString 18
- параметр TEMPDIR 19
- параметр TEMPLATEDIR 19
- параметр USEEXECUTABLE 21, 23
- параметр VIEWNOTES 36
- параметры
 - ADDEXTENSION 30
 - ADDFIELDSTODOCID 30
 - ADDNOTES 31
 - AFPVIEWING 31
 - APPLETCACHEDIR 13
 - APPLETDIR 13
 - AUTODOCRETRIEVAL 32
 - BEGIN 29
 - CACHEDIR 14
 - CACHEDOCS 14
 - CACHEMAXTHRESHOLD 15
 - CACHEMINTHRESHOLD 15
 - CACHESIZE 15
 - CACHEUSERIDS 16
 - CODEPAGE 16
 - CONFIGFILE 21, 22
 - DOCSIZE 16
 - EMAILVIEWING 32
 - ENCRYPTCOOKIES 33
 - ENCRYPTURL 33
 - END 29
 - FOLDERDESC 33
 - HOST 12
 - IMAGEDIR 17
 - INSTALLDIR 21, 23
 - LANGUAGE 18
 - LINEVIEWING 34
 - LOG 38, 143
 - LOGDIR 38
 - MAXHITS 34
 - NOLINKS 35
 - ODApplet.jre.path.IE 35
 - ODApplet.jre.path.NN 35
 - ODApplet.version 35
 - PORT 11, 12
 - PROTOCOL 12, 13
 - REPORTSERVERTIMEOUT 20
 - SEPARATOR 29
 - SERVERACCESS 20
 - SERVERPRINT 35
 - SERVERPRINTERS 36
 - SHOWDOCLOCATION 36

- параметры *(продолжение)*
 - ShowSearchString 18
 - TEMPDIR 19
 - TEMPLATEDIR 19
 - USEEXECUTABLE 21, 23
 - VIEWNOTES 36
 - раздел @SRV@_DEFAULT 11
 - раздел @SRV@_server 12
 - раздел AFP2HTML 20
 - раздел AFP2PDF 22
 - раздел ATTACHMENT IMAGES 27
 - раздел CONFIGURATION 13
 - раздел SECURITY 19
 - пароли
 - API Java 96, 97, 127
 - переменные среды, API Java 92
 - печатать документ
 - API 73
 - API Java 119
 - описание функции 7
 - параметры 73
 - пример вызова функции 76
 - печать
 - API Java 119
 - сервер 35, 36
 - подготовка к работе с соединением Internet OnDemand 1
 - поддержка языков 141
 - подключение к серверу 96, 97
 - поиск в папке 105, 107, 109, 114
 - поиск папки 100
 - поисковые значения, API Java 100
 - показать расположение документа 36
 - пользовательские файлы
 - установка 47
 - порт
 - API Java 97
 - права доступа 9
 - права доступа к каталогу 9
 - преобразование шрифтов AFP 51
 - прикладной программный интерфейс (API)
 - См. API
 - пример исходного кода 91
 - примеры 91
 - примеры приложений 41
 - программа просмотра строковых данных
 - настройка 52
 - параметр ODApplet.jre.path.IE 35
 - параметр ODApplet.jre.path.NN 35
 - параметр ODApplet.jre.version 35
 - параметр ODApplet.version 35
 - программа просмотра строковых данных Java
 - настройка 52
 - параметр ODApplet.jre.path.IE 35
 - параметр ODApplet.jre.path.NN 35
 - параметр ODApplet.jre.version 35
 - параметр ODApplet.version 35
 - программное обеспечение ODWEK
 - установка 9
 - программное обеспечение сервера HTTP
 - установка 9
 - просмотреть комментарии
 - API 85
 - вывод ASCII с ограничителями 140
 - описание функции 7
 - параметры 85
 - пример вызова функции 86
 - протокол связи TCP/IP 12, 13
 - протоколы 12, 13
 - протоколы связи 12, 13
- ## P
- раздел @SRV@_DEFAULT 11
 - раздел @SRV@_server 12
 - Раздел AFP2HTML 20
 - раздел AFP2PDF 22
 - раздел ATTACHMENT IMAGES 27
 - раздел browser 36
 - раздел CONFIGURATION 13
 - раздел debug 37
 - раздел DEFAULT BROWSER 30
 - раздел MIMETYPES 23
 - раздел NOHTML 29
 - раздел SECURITY 19
 - размер кэша 15
 - расположение документа 36
 - результаты запроса 34
 - руководство программиста
 - API 91
 - API Java 91
- ## C
- сервер
 - API Java 96, 97
 - сервер HTTP
 - файл httpd.conf 135
 - файлы конфигурации 135
 - сервер HTTP Apache
 - файлы конфигурации 135
 - сервер, печать
 - API 73
 - API Java 119
 - описание функции 7
 - параметры 73
 - пример вызова функции 76
 - разрешение применения 35, 36
 - сервлет
 - справочник 87
 - сервлет Java
 - справочник 87
 - соединение Internet OnDemand
 - о программе 1
 - сообщение об ошибке
 - вывод ASCII с ограничителями 140
 - сообщения 18
 - вывод ASCII с ограничителями 140
 - список доступа к серверам 20
 - список найденных документов
 - API 65
 - API Java 100, 105, 114, 116
 - вывод ASCII с ограничителями 139
 - описание функции 6
 - параметры 65
 - пример вызова функции 68
 - справка 143
 - справочник
 - API 59, 87, 89
 - API Java 89
 - CGI API 59
 - сервлет 87
 - сервлет Java 87
 - среда системы, API Java 92

ссылки 35
строка поиска SQL с API Java 105
структура пакета, Java 91

Т

тайм-аут 20
тайм-аут простоя 20
тип данных MIME 23, 100
тип документа, API Java 100
тип соединения
API Java 97
тип/подтип носителя 23
трассировка и информация диагностики 94
трассировка неполадок 143
требования
AFP2PDF Transform 10
AFP2WEB Transform 10
API Java 1
cookie 46
апплет AFP2HTML Java 10
Виртуальная машина Java 46
кэш документов 9
кэш-память 9
сервер 9
сервер HTTP 9

У

установка
AFP Web Viewer 45
Image Web Viewer 45
OS/400 10
апплет AFP2HTML Java 45
апплет строковых данных Java 45
апплеты 45
апплеты Java 45
встраиваемые модули 45
настройка 47
пользовательские файлы 47
программное обеспечение ODWEK 9
рабочая станция пользователя 45
сервер HTTP 9
требования 9
файл ARSWWW.INI 11

Ф

файл ARSWWW.INI
AFP2PDF Transform 22
AFP2WEB Transform 20
настройка 11
опции браузера 36
параметр ADDEXTENSION 30
параметр ADDFIELDSTODOCID 30
параметр ADDNOTES 31
параметр AFPVIEWING 31
параметр APPLETT 13
параметр APPLETCACHEDIR 13
параметр AUTODOCRETRIEVAL 32
параметр BEGIN 29
параметр CACHEDIR 14
параметр CACHEDOCS 14
параметр CACHEMAXTHRESHOLD 15
параметр CACHEMINTHRESHOLD 15
параметр CACHESIZE 15

файл ARSWWW.INI (продолжение)
параметр CACHEUSERIDS 16
параметр CODEPAGE 16
параметр CONFIGFILE 21, 22
параметр DOCSIZE 16
параметр EMAILVIEWING 32
параметр ENCRYPTCOOKIES 33
параметр ENCRYPTURL 33
параметр END 29
параметр FOLDERDESC 33
параметр HOST 12
параметр IMAGEDIR 17
параметр INSTALLDIR 21, 23
параметр LANGUAGE 18
параметр LINEVIEWING 34
параметр LOG 38, 143
параметр LOGDIR 38
параметр MAXHITS 34
параметр NOLINKS 35
параметр ODApplet.jre.path.IE 35
параметр ODApplet.jre.path.NN 35
параметр ODApplet.jre.version 35
параметр ODApplet.version 35
параметр PORT 11, 12
параметр PROTOCOL 12, 13
параметр REPORTSERVERTIMEOUT 20
параметр SEPARATOR 29
параметр SERVERACCESS 20
параметр SERVERPRINT 35
параметр SERVERPRINTERS 36
параметр SHOWDOCLOCATION 36
параметр ShowSearchString 18
параметр TEMPDIR 19
параметр TEMPLATEDIR 19
параметр USEEXECUTABLE 21, 23
параметр VIEWNOTES 36
раздел @SRV@_DEFAULT 11
раздел AFP2HTML 20
раздел AFP2PDF 22
раздел ATTACHMENT IMAGES 27
раздел browser 36
раздел CONFIGURATION 13
раздел debug 37
раздел DEFAULT BROWSER 30
раздел MIMETYPES 23
раздел NOHTML 29
раздел SECURITY 19
указание 11
файл httpd.conf 135
файл конфигурации AFP2HTML 129
файл конфигурации AFP2PDF 133
файл шаблона 42
файлы протоколов 38, 143
функции
вход в систему 6
выход из системы 6
добавить комментарий 6
загрузить документ 6
изменить пароль 6
комментарии 6, 7
критерии поиска 7
обновить документ 7
печатать документ 7
печать документа на сервере 7
просмотреть комментарии 7
список найденных документов 6

Ш

шифрование 33

шрифты

 AFP 51

 TrueType 51

 преобразование 51

шрифты AFP

 преобразование 51

шрифты TrueType

 преобразование шрифтов AFP в 51

Я

язык 18, 141

языки набора двухбайтовых символов 141



Номер программы: 5722-RD1

SC43-0361-02

